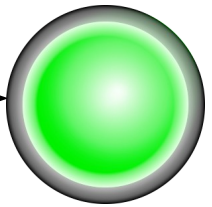
Deep Learning & Convolutional Networks

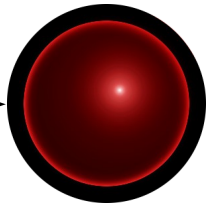
Yann LeCun
Facebook AI Research &
Center for Data Science, NYU
yann@cs.nyu.edu
http://yann.lecun.com

Machine Learning (Supervised Lerning)

- training a machine to distinguish cars from airplanes.
- We show it lots of examples of cars and airplanes
- Each time, we adjust the “knobs” so it produces a good answer



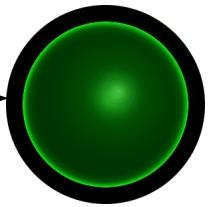
PLANE



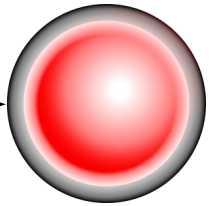
CAR

Machine Learning

- training a machine to distinguish cars from airplanes.
- We show it lots of examples of cars and airplanes
- Each time, we adjust the “knobs” so it produces a good answer



PLANE



CAR

Large-Scale Machine Learning: the reality

- Hundreds of millions of knobs
- Thousands of categories
- Millions of training samples
- Recognizing each sample may take billions of operations
  - But these operations are simple multiplications and additions

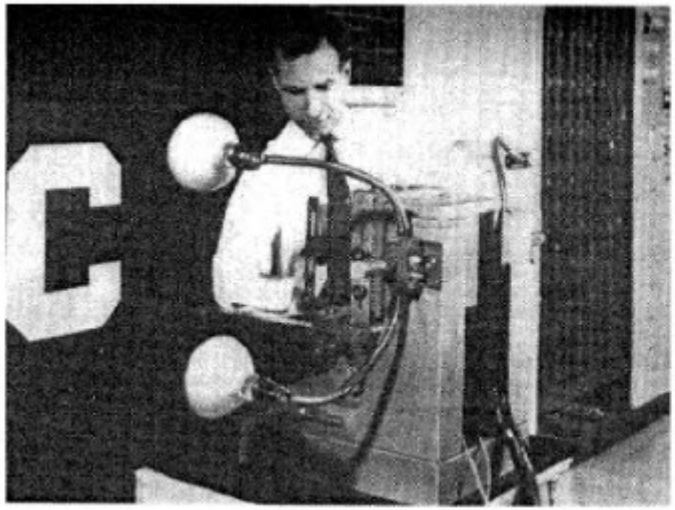
55 years of hand-crafted features

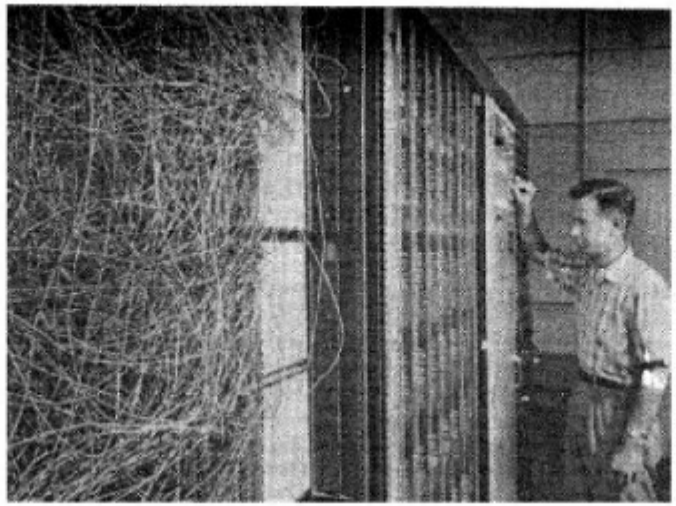
The traditional model of pattern recognition (since the late 50's)

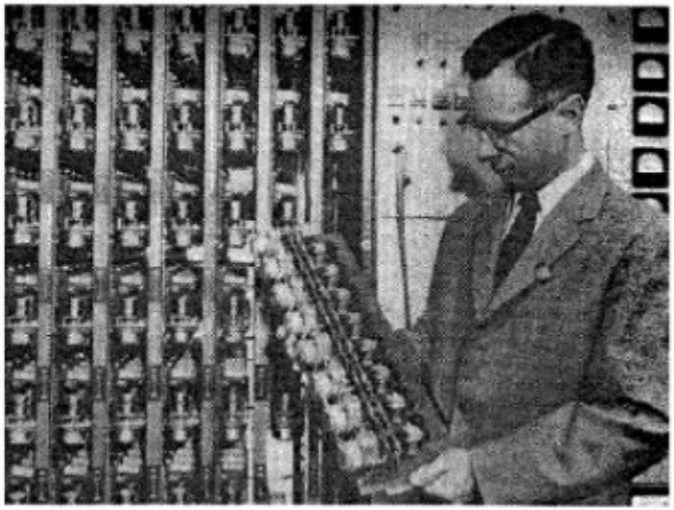
- Fixed/engineered features (or fixed kernel) + trainable classifier

hand-crafted Feature Extractor → "Simple" Trainable Classifier →

Perceptron







Deep Learning = Learning Hierarchical Representations

Y LeCun

- **Traditional Pattern Recognition:** Fixed/Handcrafted Feature Extractor

Feature Extractor → Trainable Classifier

- **Mainstream Modern Pattern Recognition:** Unsupervised mid-level features

Feature Extractor → Mid-Level Features → Trainable Classifier

- **Deep Learning:** Representations are hierarchical and trained

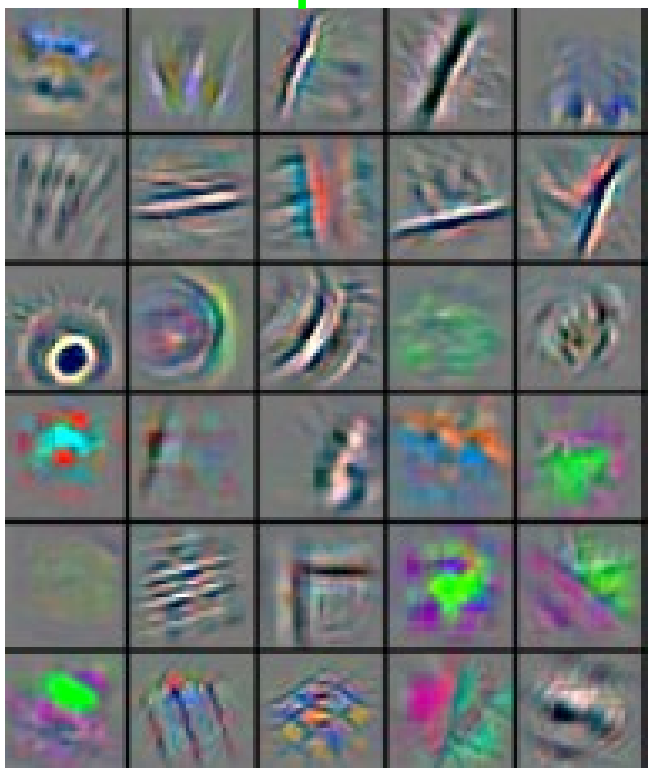
Low-Level Features → Mid-Level Features → High-Level Features → Trainable Classifier

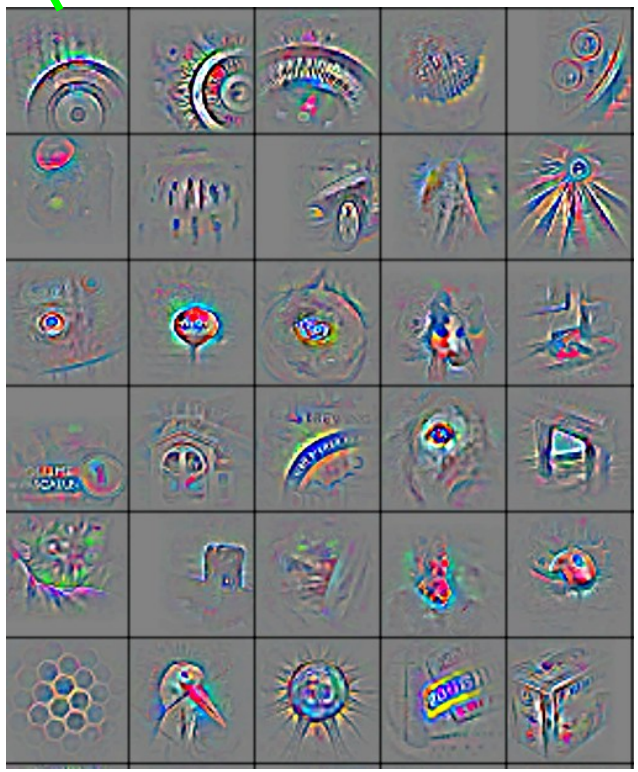
Deep Learning = Learning Hierarchical Representations

Y LeCun

- It's deep if it has more than one stage of non-linear feature transformation

Low-Level Feature → Mid-Level Feature → High-Level Feature → Trainable Classifier

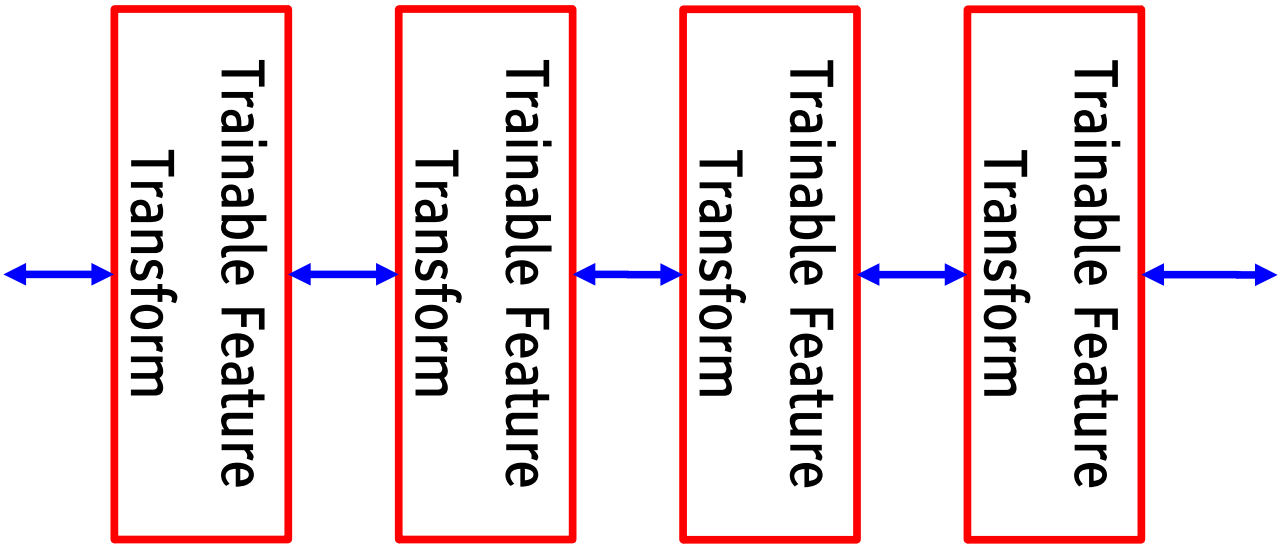




Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

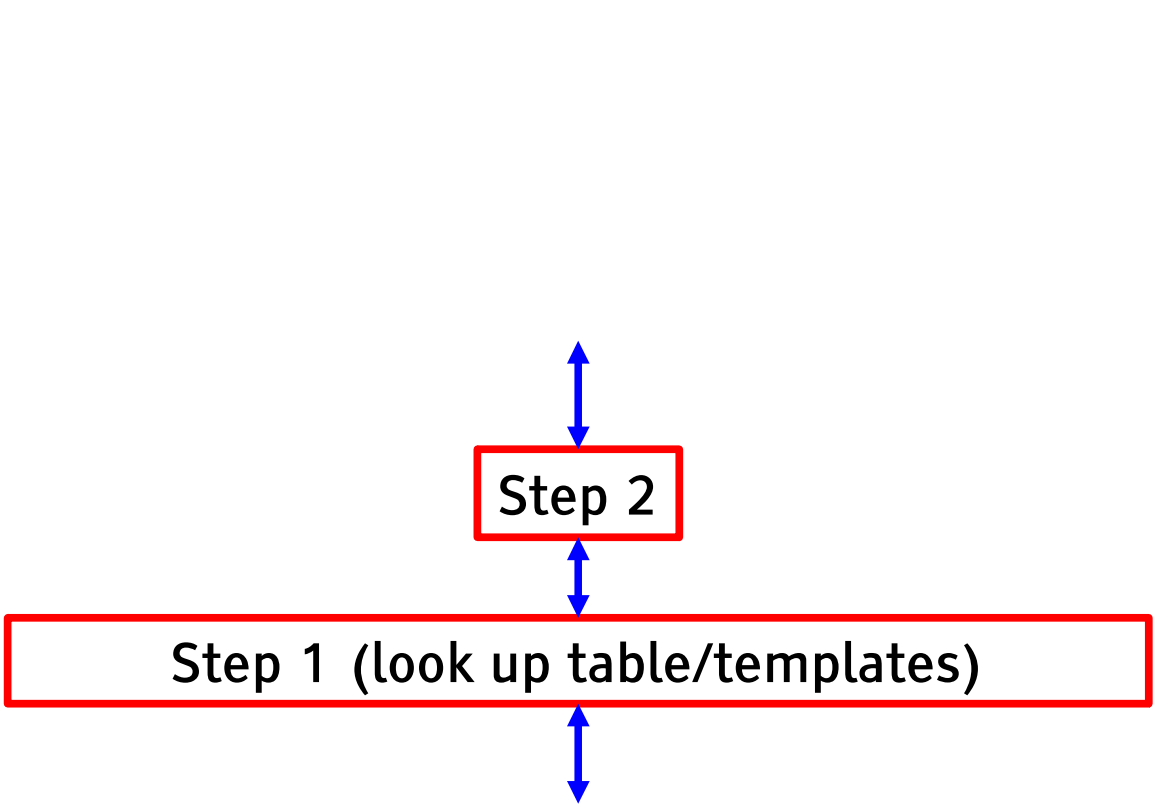
Trainable Feature Hierarchy

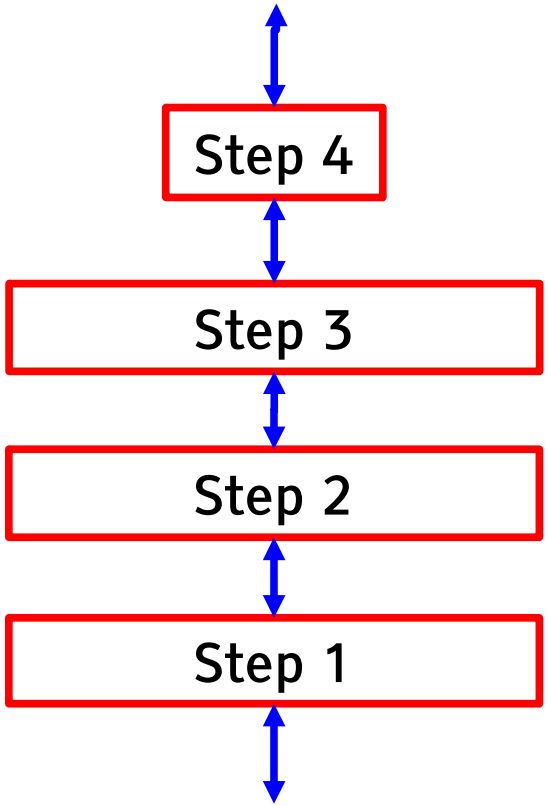
- Hierarchy of representations with increasing level of abstraction
- Each stage is a kind of trainable feature transform
- Image recognition
  - Pixel → edge → texton → motif → part → object
- Text
  - Character → word → word group → clause → sentence → story
- Speech
  - Sample → spectral band → sound → … → phone → phoneme → word



Shallow vs Deep == lookup table vs multi-step algorithm

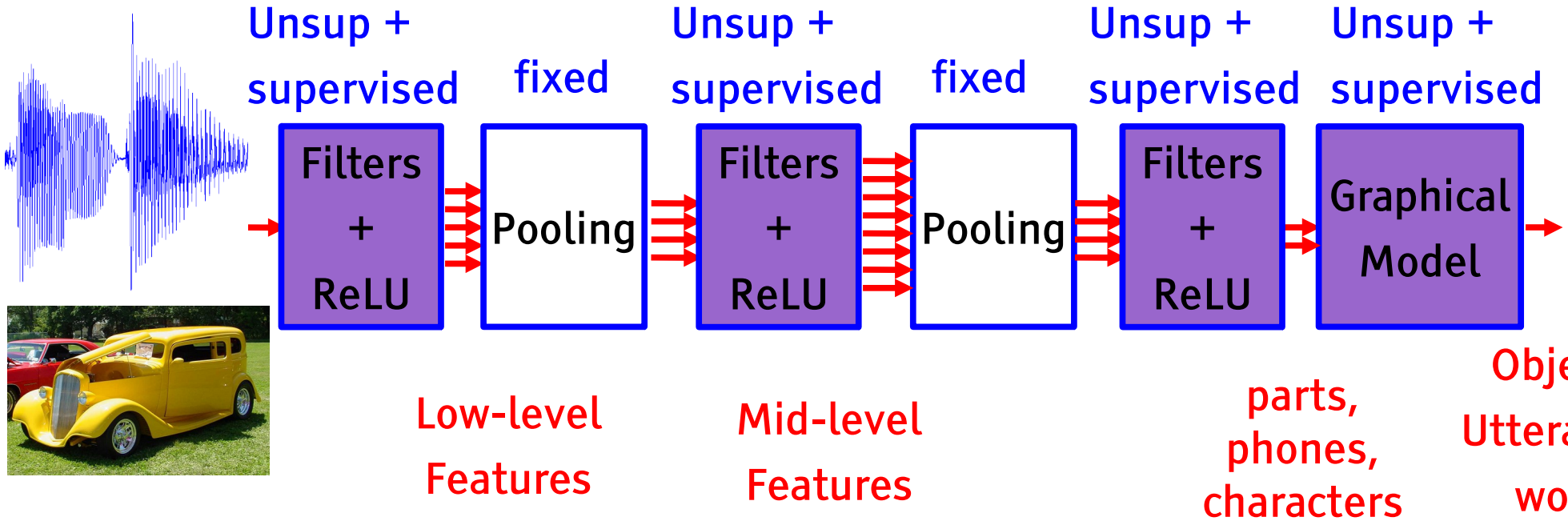
- "shallow & wide" vs "deep and narrow" == "more memory" vs "more time"
  - Look-up table vs algorithm
  - Few functions can be computed in two steps without an exponentially large lookup table
  - Using more than 2 steps can reduce the "memory" by an exponential factor.





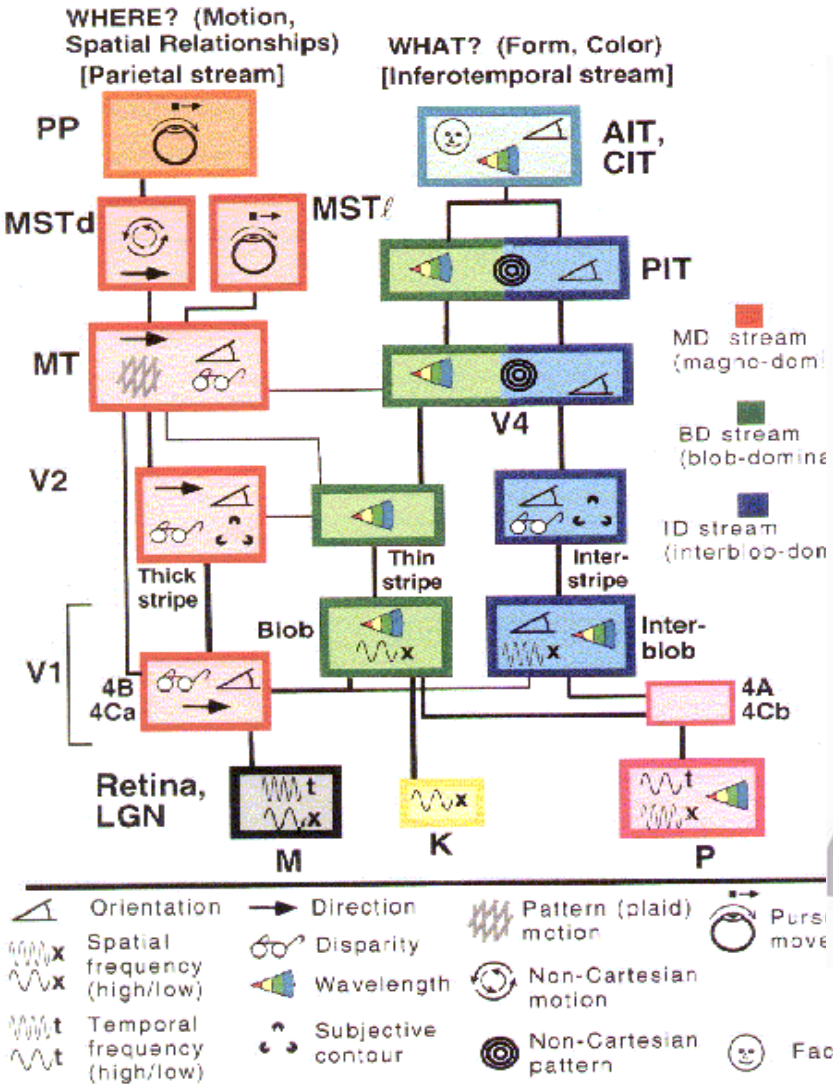
Current Research: deep learning + structured prediction

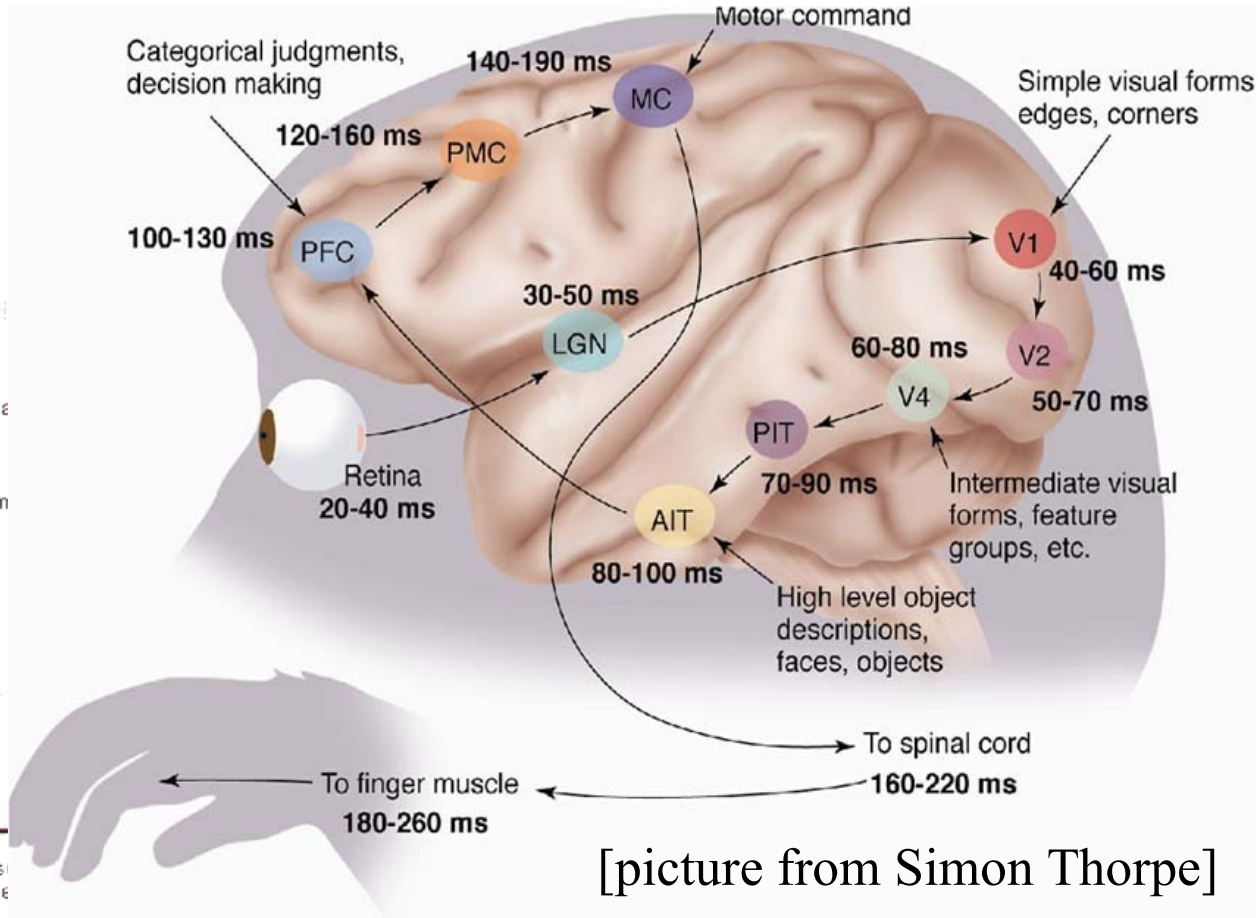
- Globally-trained deep architecture
  - Handwriting recognition: since the mid 1990s
  - Speech Recognition: since 2011
  - All the modules are trained with a combination of unsupervised and supervised learning
  - **End-to-end training == deep structured prediction**



How does the brain interprets images?

- The ventral (recognition) pathway in the visual cortex has multiple stages
- Retina - LGN - V1 - V2 - V4 - PIT - AIT ....





[picture from Simon Thorpe]

[Gallant & Van Essen]

Let's be inspired by nature, but not too much

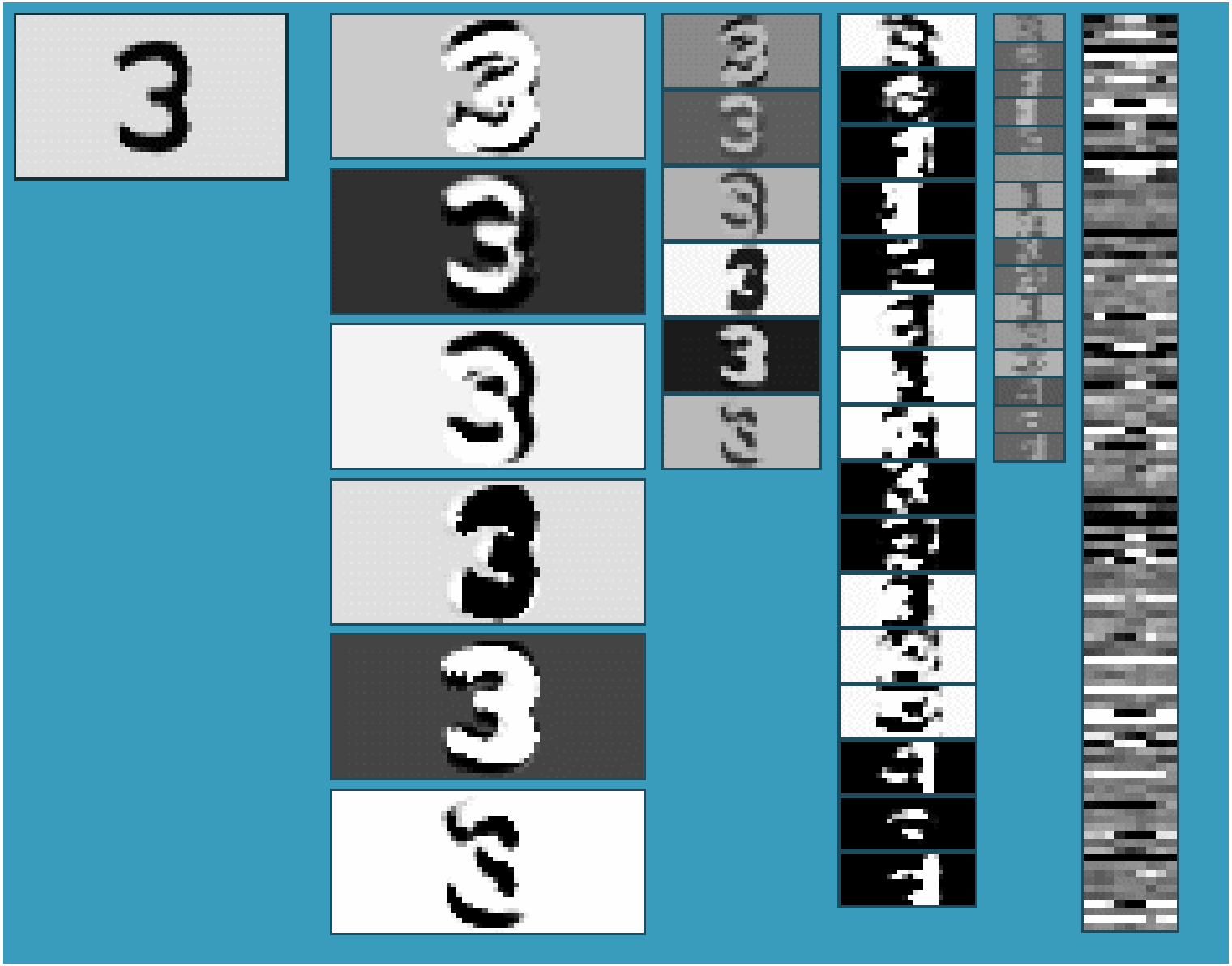
Y LeCun

- It's nice imitate Nature,
- But we also need to understand
  - How do we know which details are important?
  - Which details are merely the result of evolution, and the constraints of biochemistry?
- For airplanes, we developed aerodynamics and compressible fluid dynamics.
  - We figured that feathers and wing flapping weren't crucial
- QUESTION: What is the equivalent of aerodynamics for understanding intelligence?

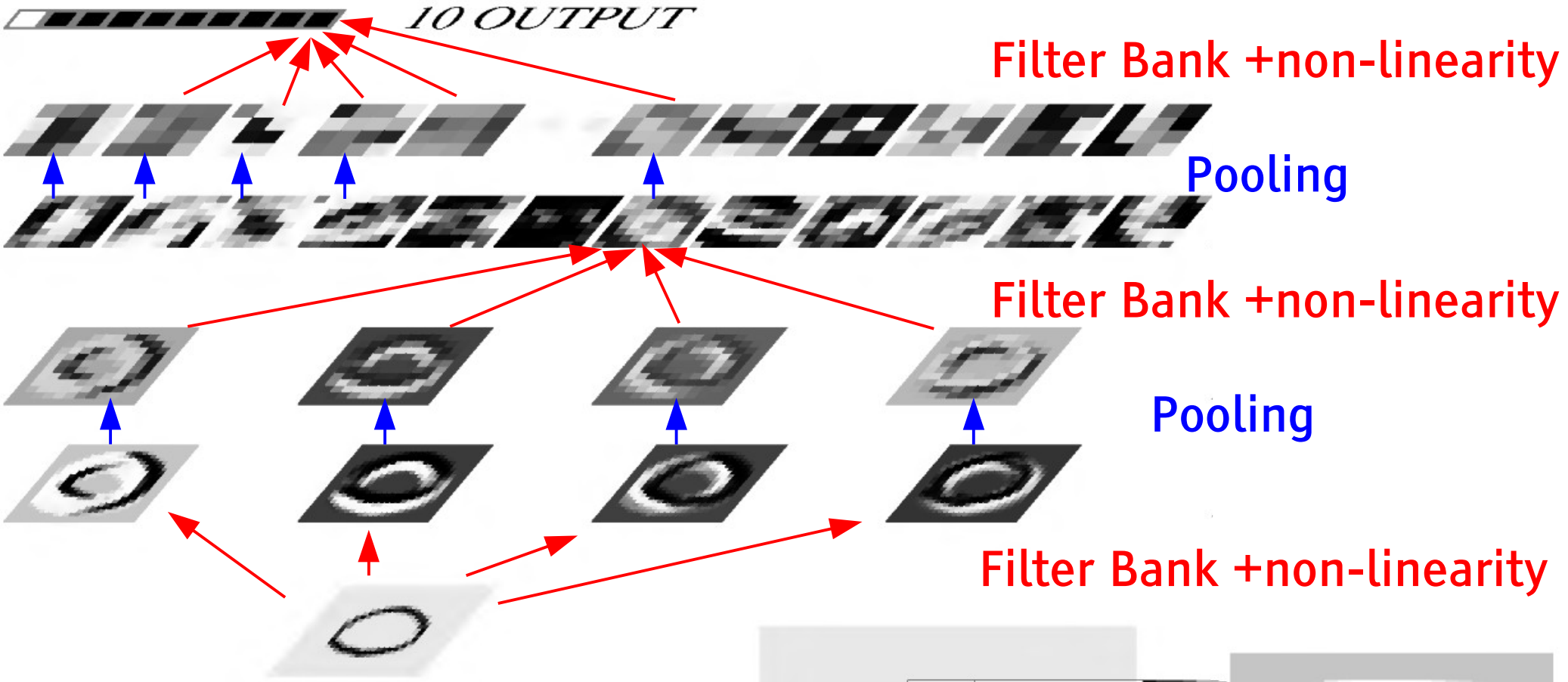
L'Avion III de Clément Ader, 1897
(Musée du CNAM, Paris)
His "Eole" took off from the ground in 1890,
13 years before the Wright Brothers, but you probably never heard of it (unless you are french).

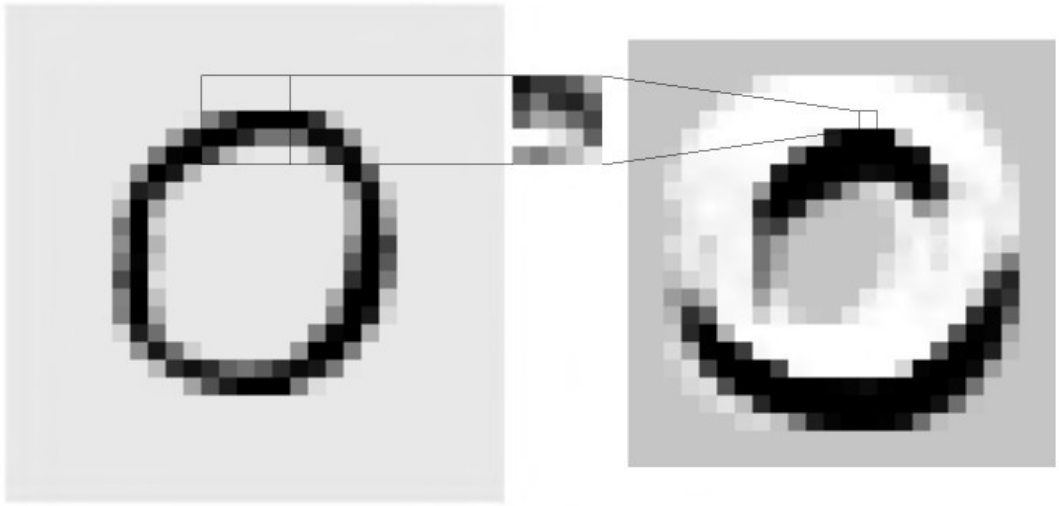
Convolutional Network (vintage 1990)

- Filters-tanh → pooling → filters-tanh → pooling → filters-tanh



Convolutional Network

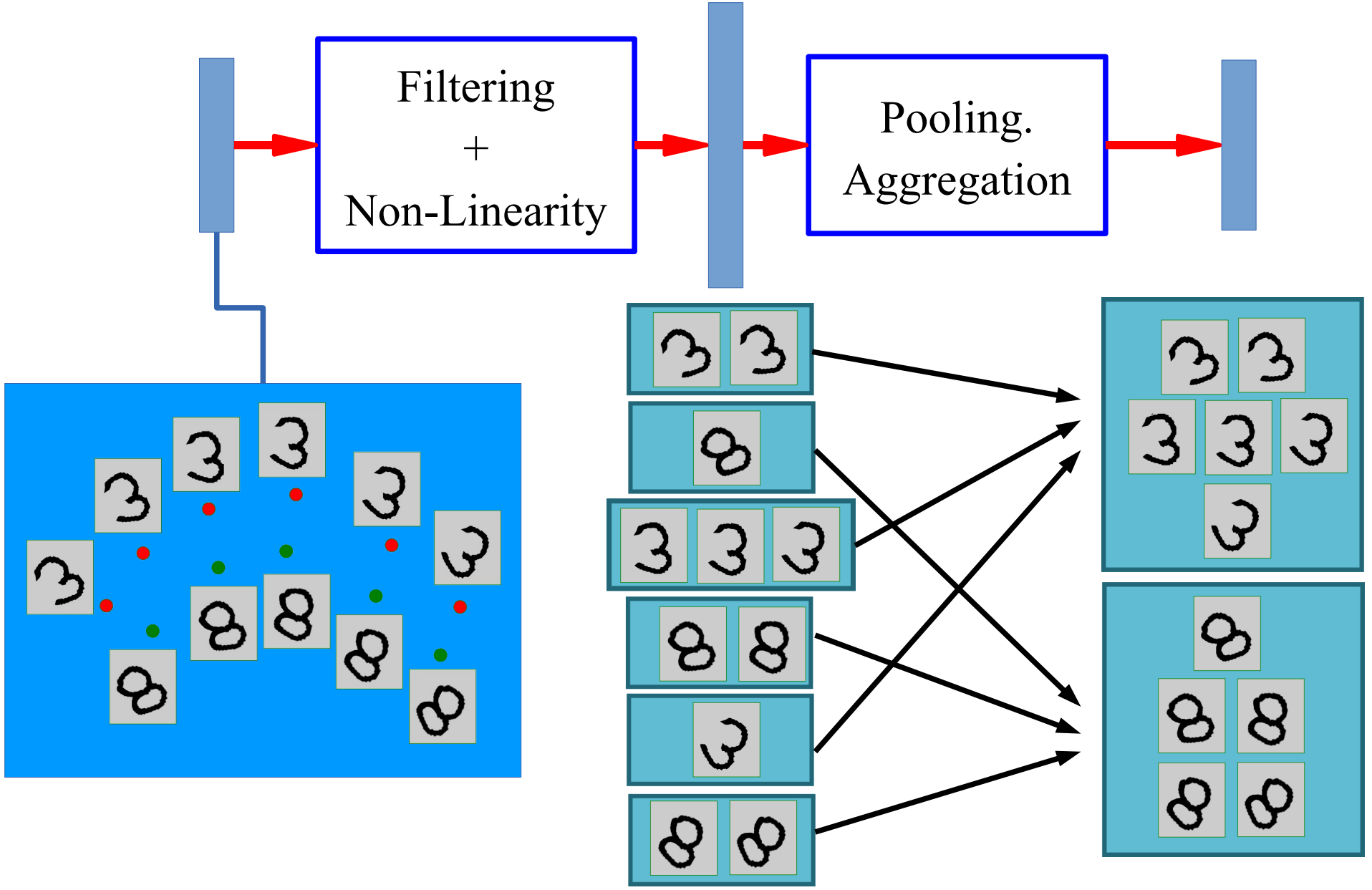




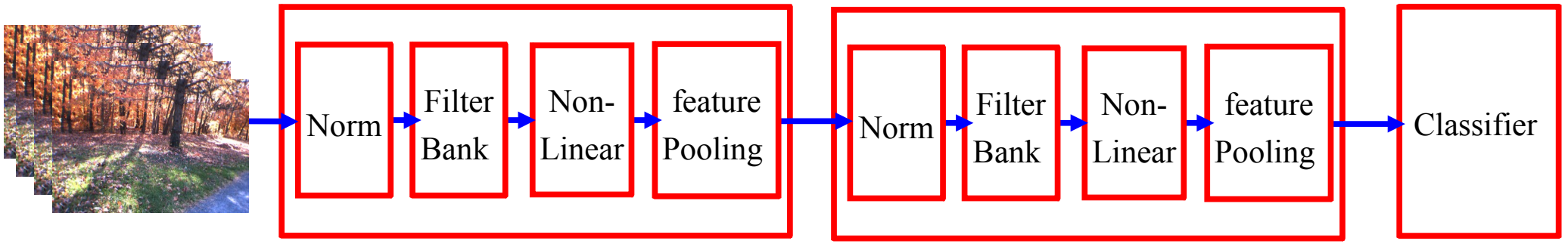
[LeCun et al. NIPS 1989]

Sparse Non-Linear Expansion → Pooling

- First, break things apart, Second pool together similar things

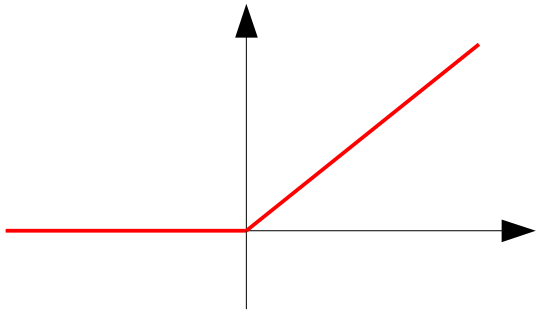


Overall Architecture: multiple stages of Normalization → Filter Bank → Non-Linearity → Pooling



- **Normalization:** variation on whitening (optional)
  - Subtractive: average removal, high pass filtering
  - Divisive: local contrast normalization, variance normalization
- **Filter Bank:** dimension expansion, projection on overcomplete basis
- **Non-Linearity:** sparsification, saturation, lateral inhibition....
  - Rectification (ReLU), Component-wise shrinkage, tanh,..

$$ReLU(x) = max(x, 0)$$

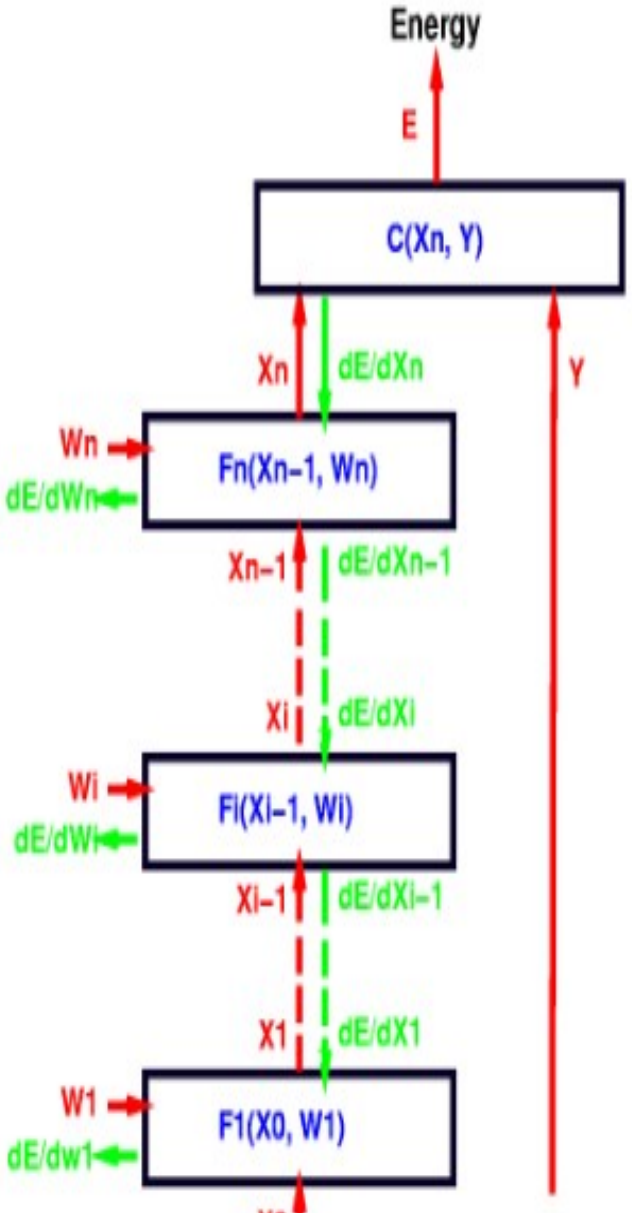


- **Pooling:** aggregation over space or feature type
  - Max, Lp norm, log prob.

$$MAX: Max_i(X_i); \quad L_p: \sqrt[p]{X_i^p}; \quad PROB: \frac{1}{b}\log\left(\sum_i e^{bX_i}\right)$$

Supervised Training: Stochastic (Sub)Gradient Optimization

algorithm that uses the recurrence equation for $\frac{\partial E}{\partial X_i}$



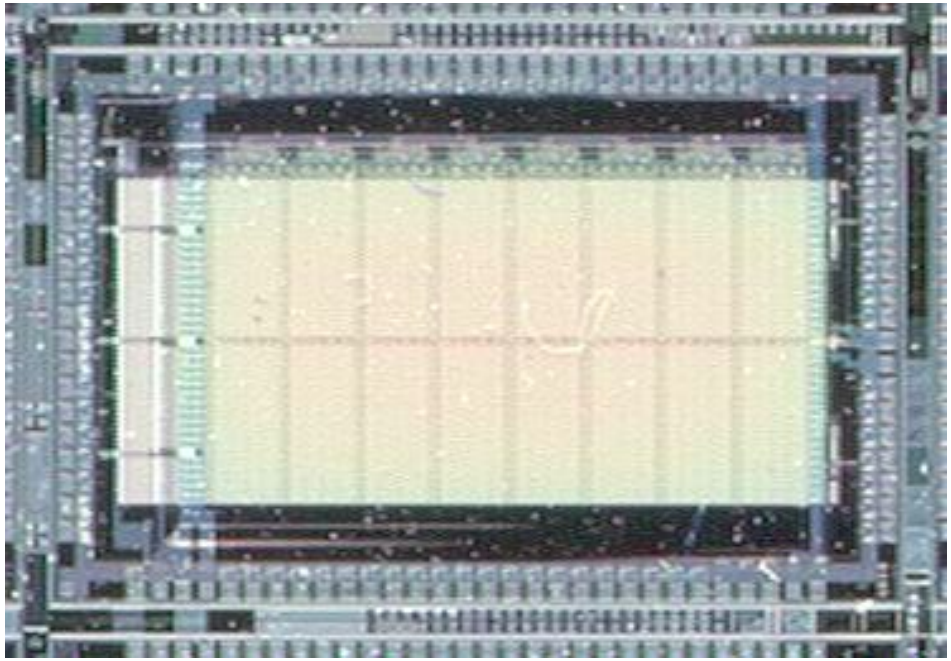
- $\frac{\partial E}{\partial X_n} = \frac{\partial C(X_n, Y)}{\partial X_n}$
- $\frac{\partial E}{\partial X_{n-1}} = \frac{\partial E}{\partial X_n} \frac{\partial F_n(X_{n-1}, W_n)}{\partial X_{n-1}}$
- $\frac{\partial E}{\partial W_n} = \frac{\partial E}{\partial X_n} \frac{\partial F_n(X_{n-1}, W_n)}{\partial W_n}$
- $\frac{\partial E}{\partial X_{n-2}} = \frac{\partial E}{\partial X_{n-1}} \frac{\partial F_{n-1}(X_{n-2}, W_{n-1})}{\partial X_{n-2}}$
- $\frac{\partial E}{\partial W_{n-1}} = \frac{\partial E}{\partial X_{n-1}} \frac{\partial F_{n-1}(X_{n-2}, W_{n-1})}{\partial W_{n-1}}$
- ....etc, until we reach the first module.
- we now have all the $\frac{\partial E}{\partial W_i}$ for $i \in [1, n]$.

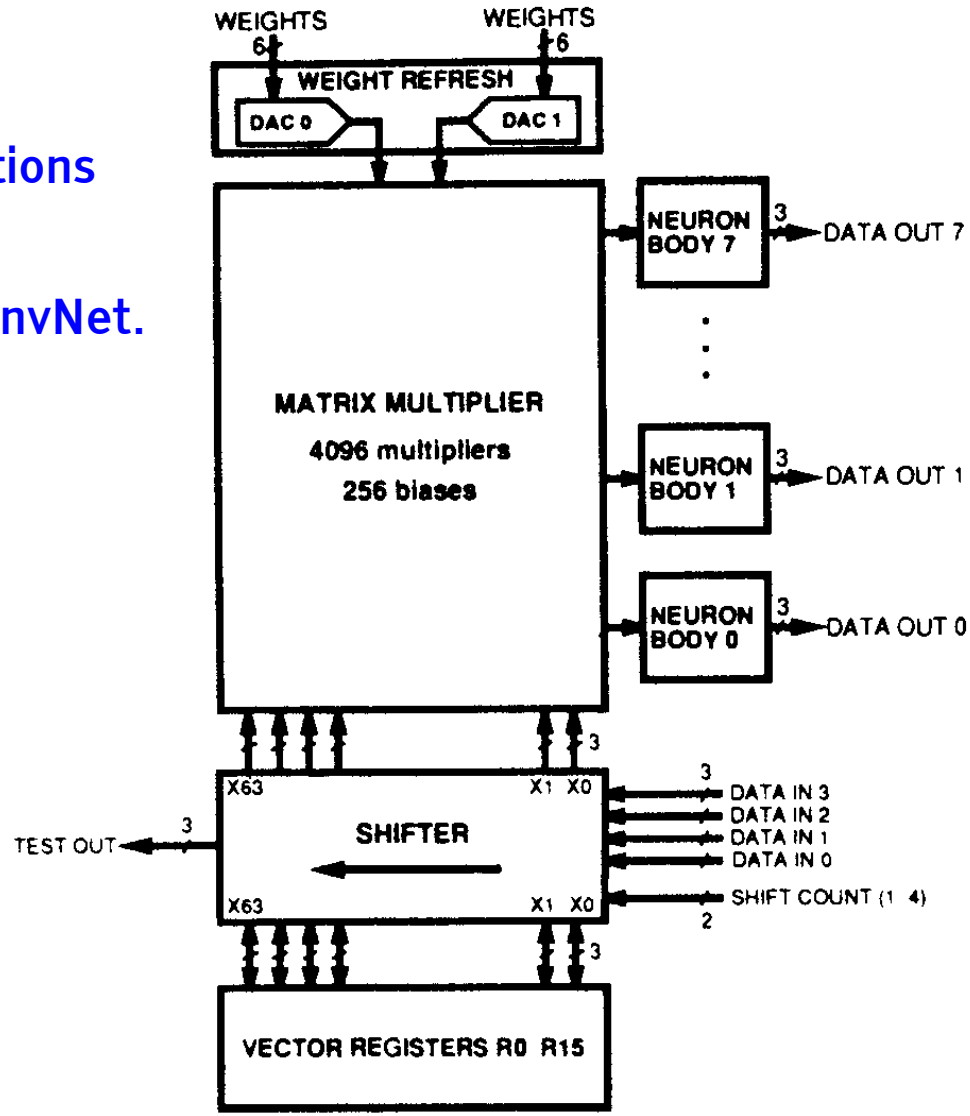
LeNet1 Demo from 1993

- Running on a 486 PC with an AT&T DSP32C add-on board (20 Mflops!)

ANNA: Analog-Digital ConvNet Accelerator Chip (Bell Labs)

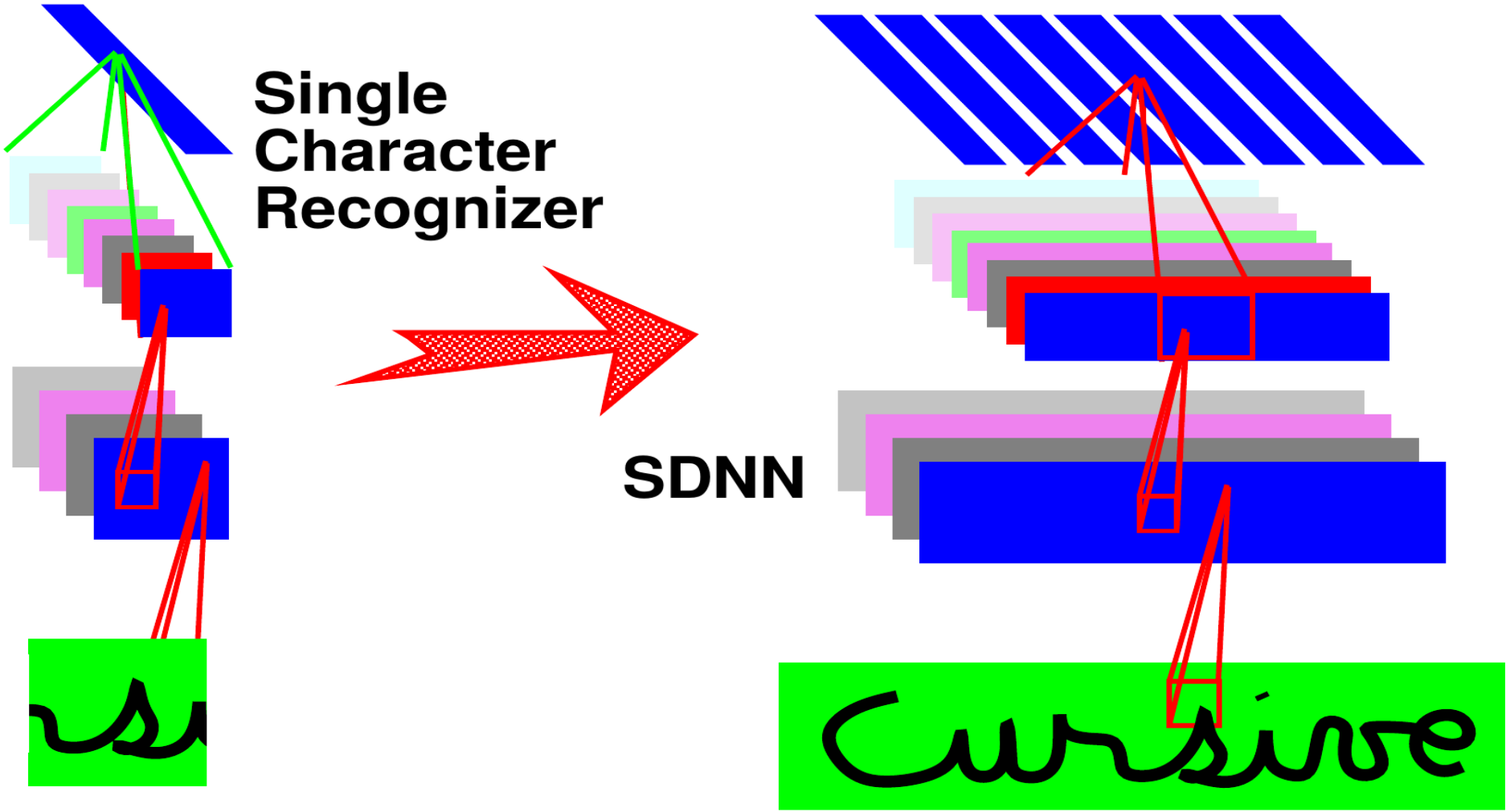
- [Boser, Säckinger, Bromley, LeCun, Jackel, IEEE J. SSC 26(12), 1991]
- 4096 Multiply-Accumulate operators
- 6 bit weights, 4 bit states
- 20 MHz clock
- Shift registers for efficient I/O with convolutions
- 4 GOPS (peak)
- 1000 characters per second for OCR with ConvNet.



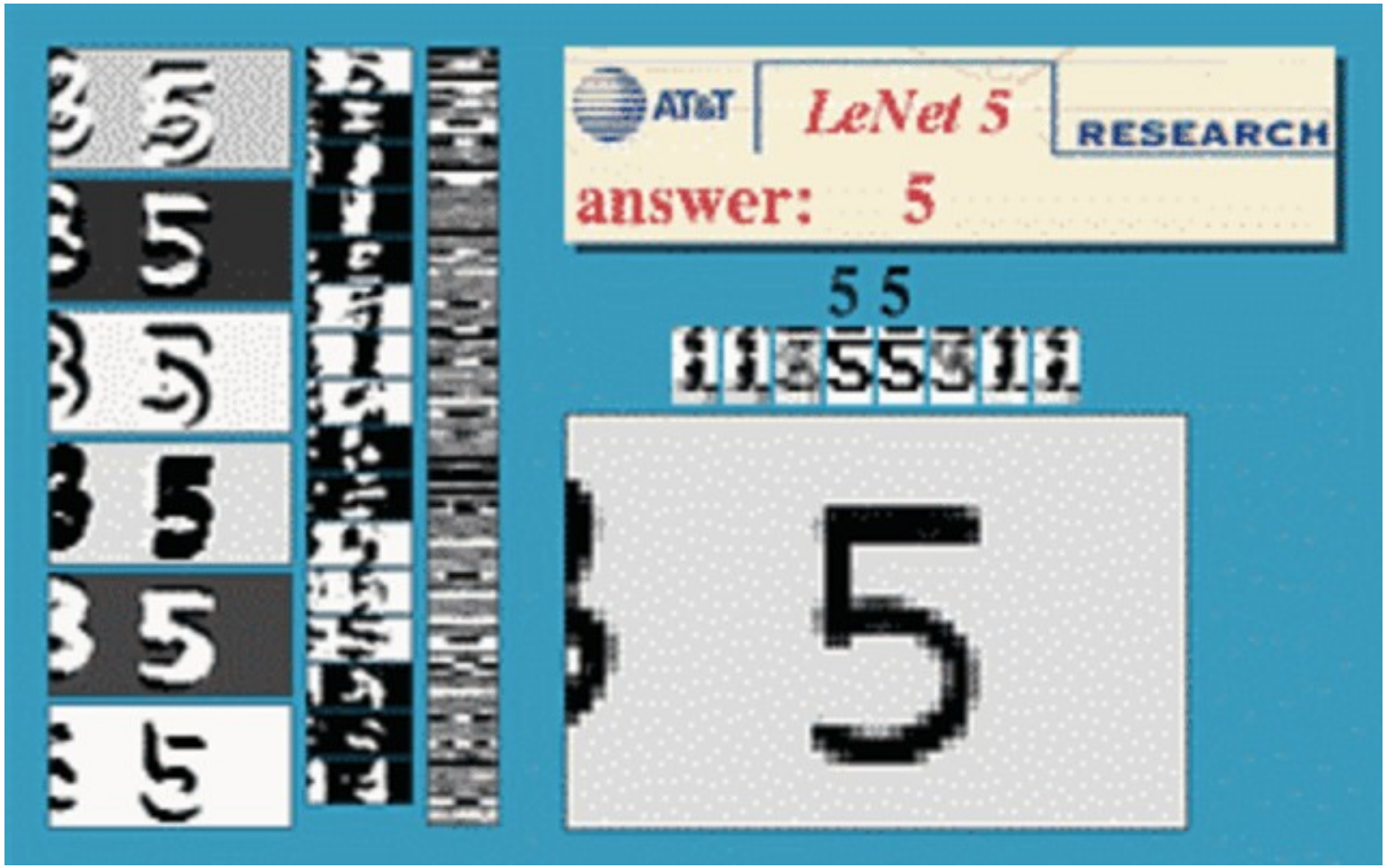


Multiple Character Recognition [Matan et al 1992]

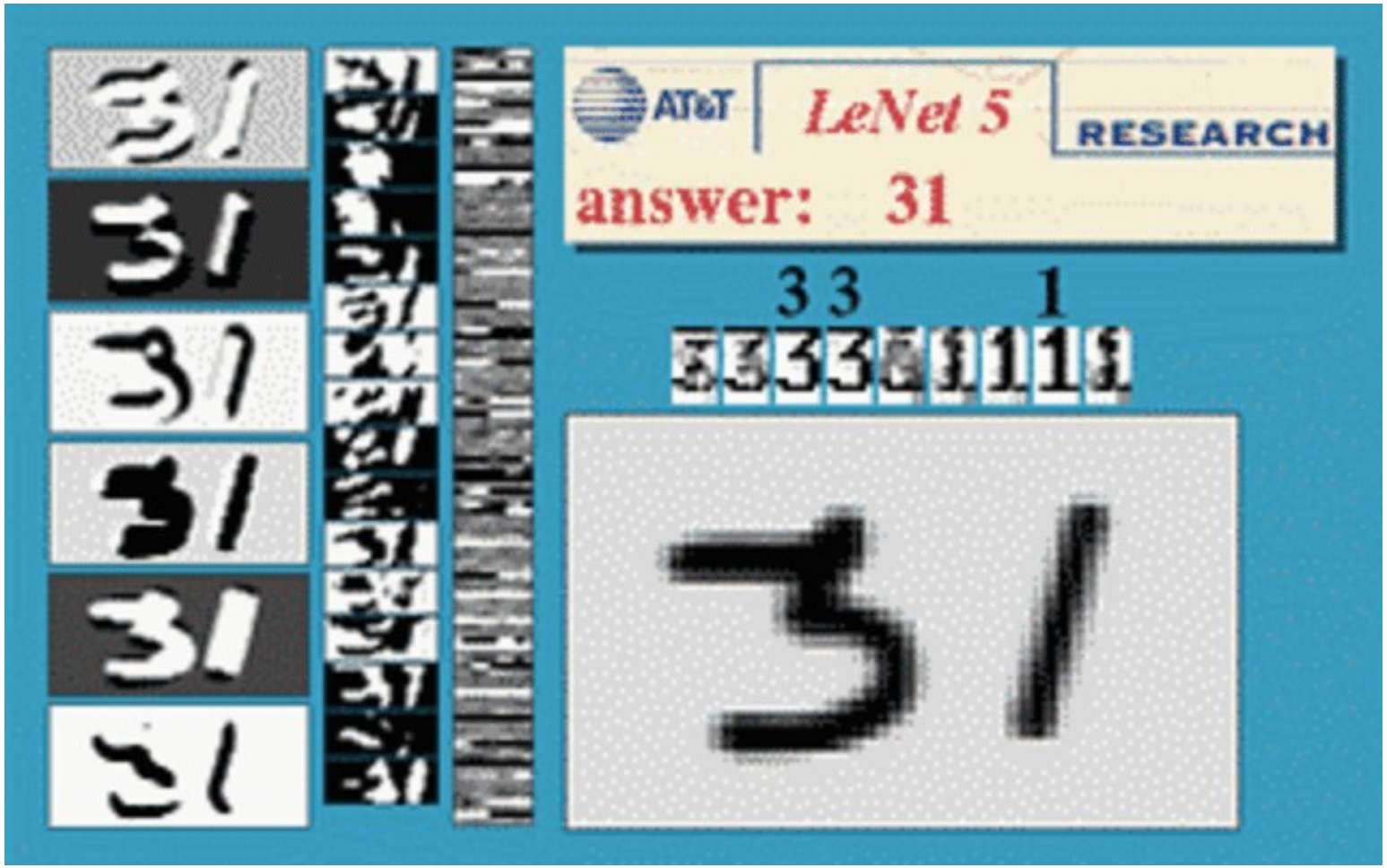
- Every layer is a convolution



Sliding Window ConvNet + Weighted Finite-State Machine



Sliding Window ConvNet + Weighted FSM



Check Reader (Bell Labs, 1995)

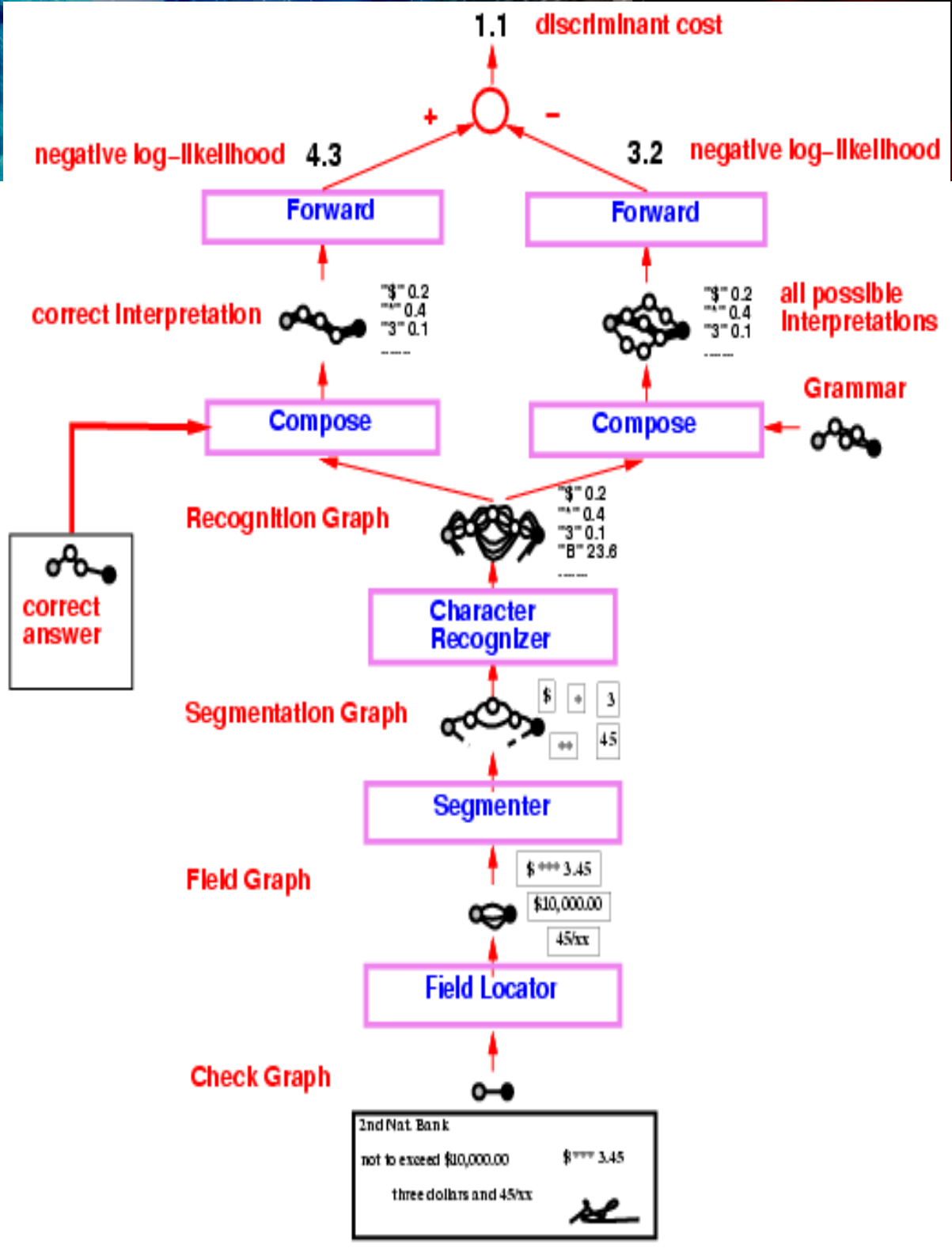
Graph transformer network trained to read check amounts.

Trained globally with Negative-Log-Likelihood loss.

50% percent correct, 49% reject, 1% error (detectable later in the process.

Fielded in 1996, used in many banks in the US and Europe.

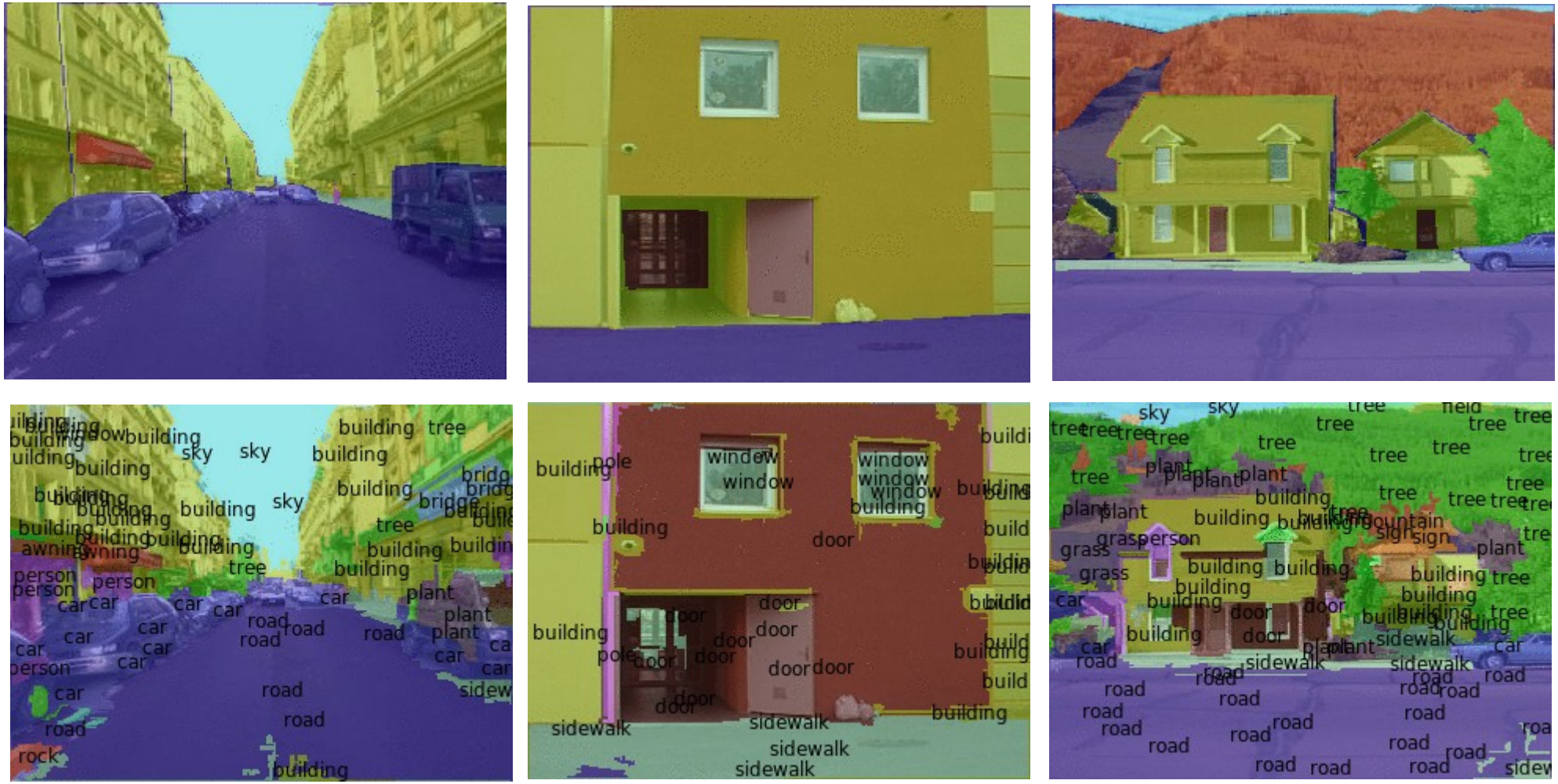
Processed an estimated 10% to 20% of all the checks written in the US in the early 2000s.



Simultaneous face detection and pose estimation

Y LeCun

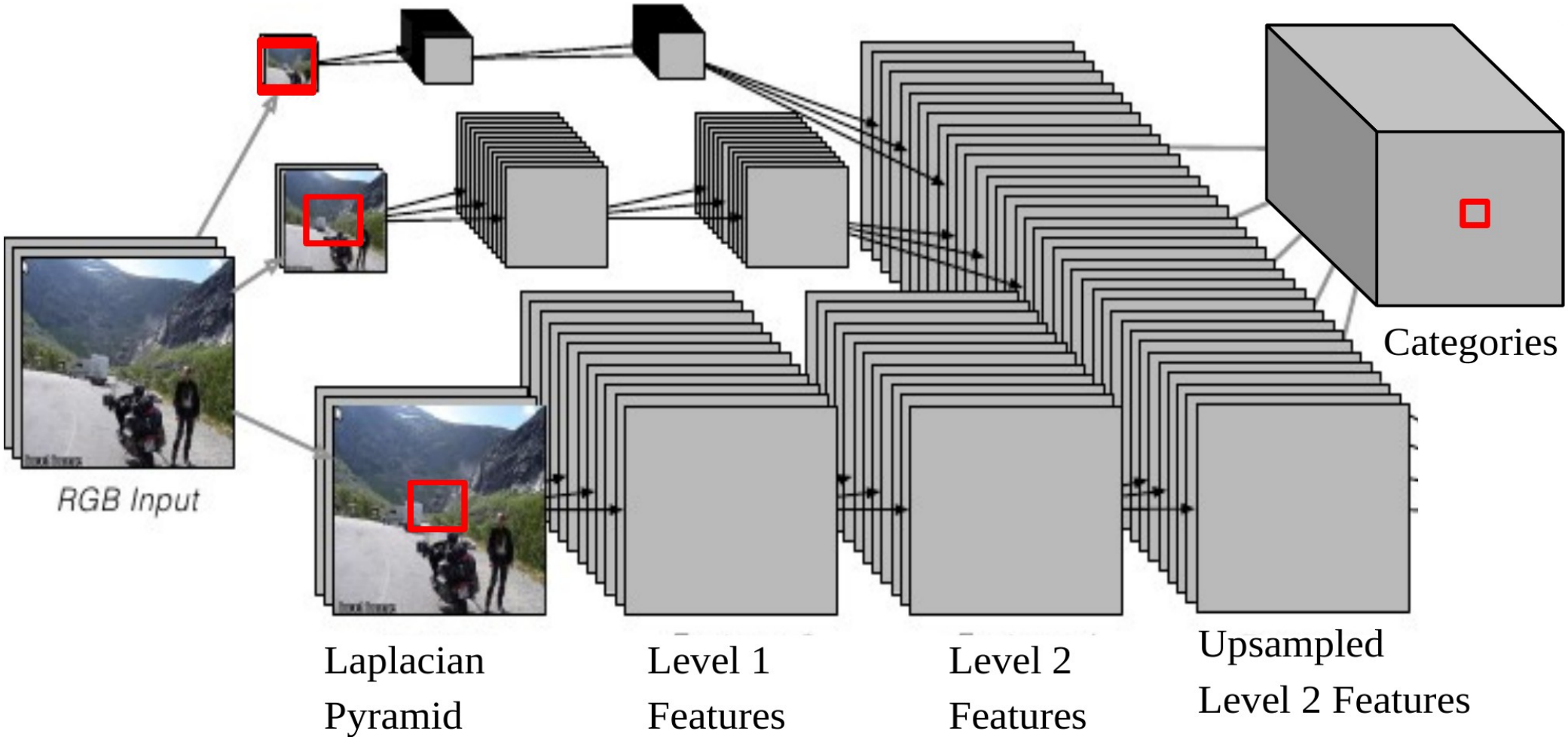
Scene Parsing/Labeling



[Farabet et al. ICML 2012, PAMI 2013]

Scene Parsing/Labeling: Multiscale ConvNet Architecture

- Each output sees a large input context:
  - **46x46** window at full rez; **92x92** at ½ rez; **184x184** at ¼ rez
  - [7x7conv]->[2x2pool]->[7x7conv]->[2x2pool]->[7x7conv]->
  - Trained supervised on fully-labeled images



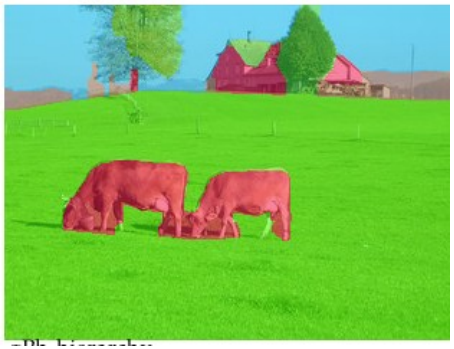
Method 1: majority over super-pixel regions

Input image

Super-pixel boundary hypetheses

Superpixel boundaries

Majority Vote Over Superpixels



Categories aligned With region boundaries

Multi-scale ConvNet

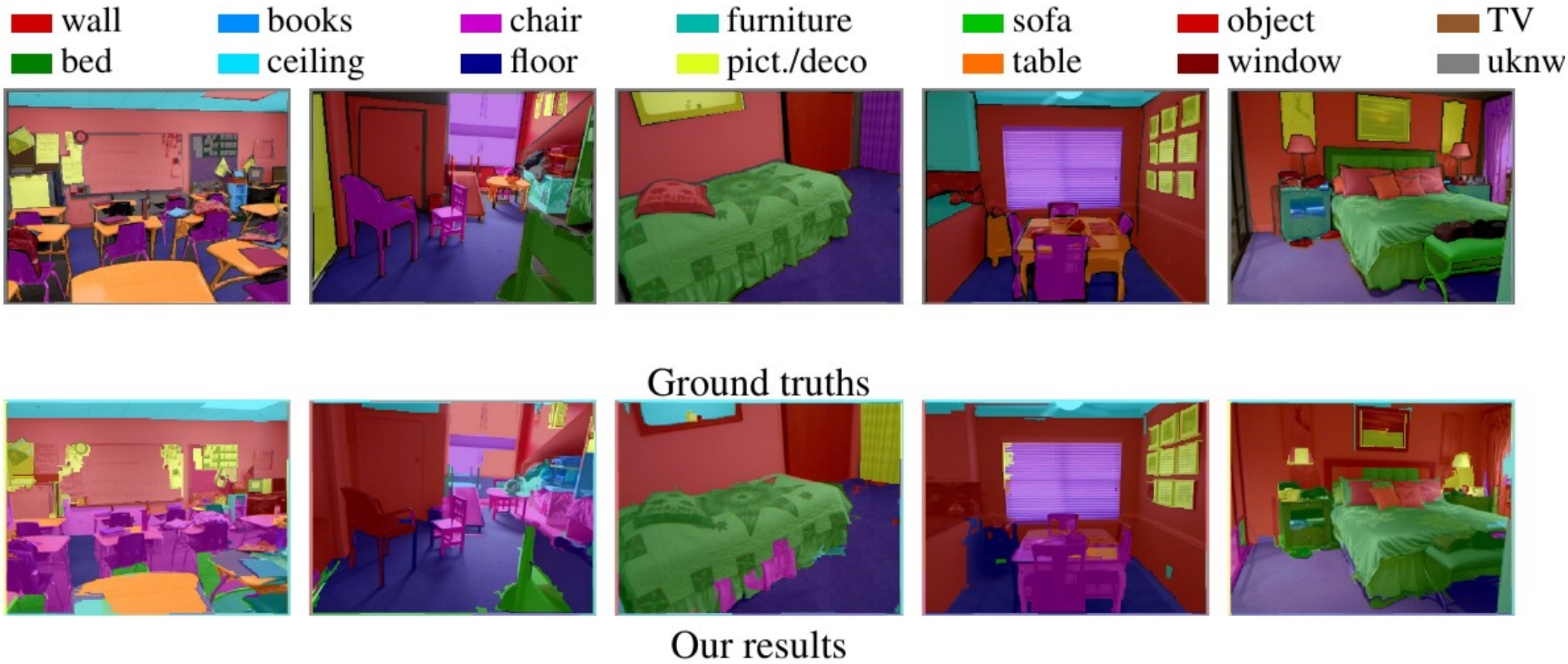
Features from Convolutional net (d=768 per pixel)

Convolutional classifier

"soft" categories scores

[Farabet et al. IEEE T. PAMI 2013]

Scene Parsing/Labeling on RGB+Depth Images

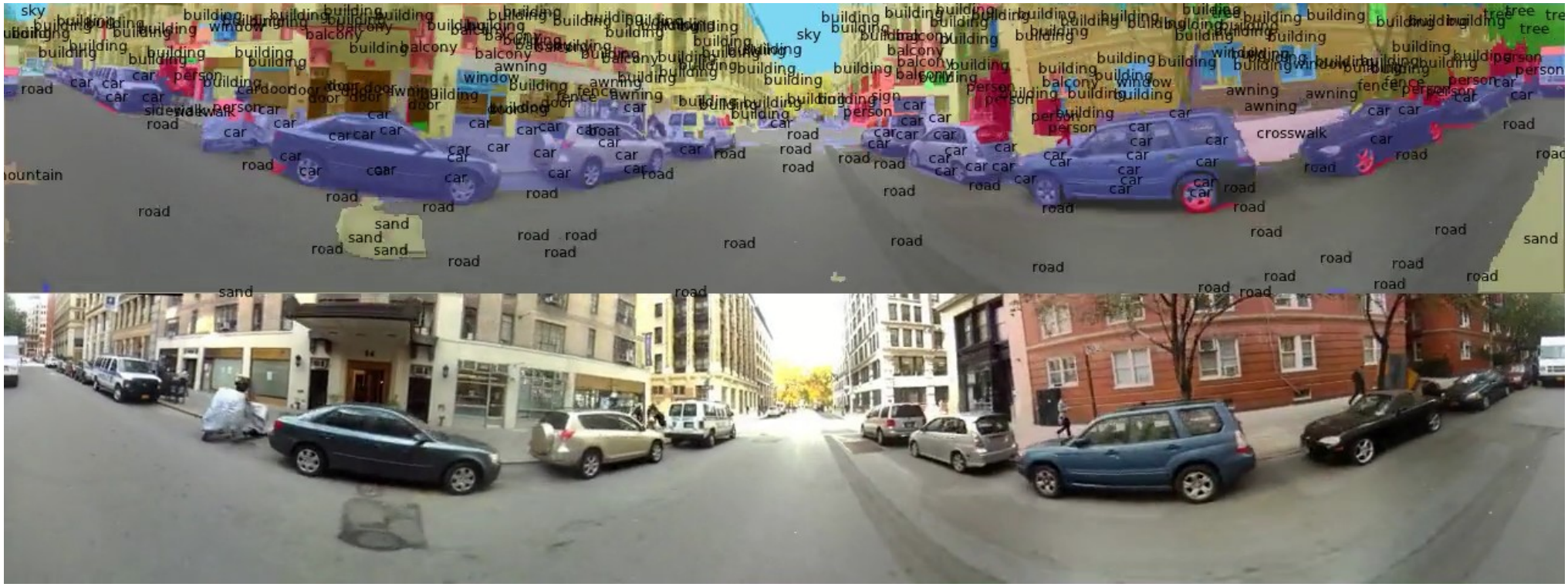


[Couprie, Farabet, Najman, LeCun ICLR 2013, ICIP 2013]

Scene Parsing/Labeling

[Farabet et al. ICML 2012, PAMI 2013]

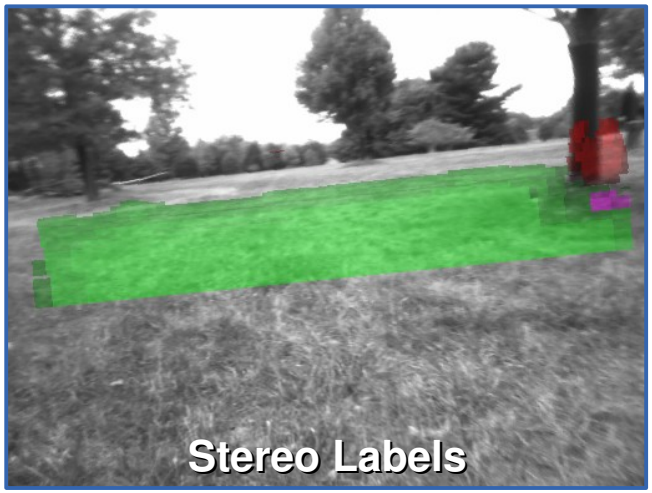
Scene Parsing/Labeling

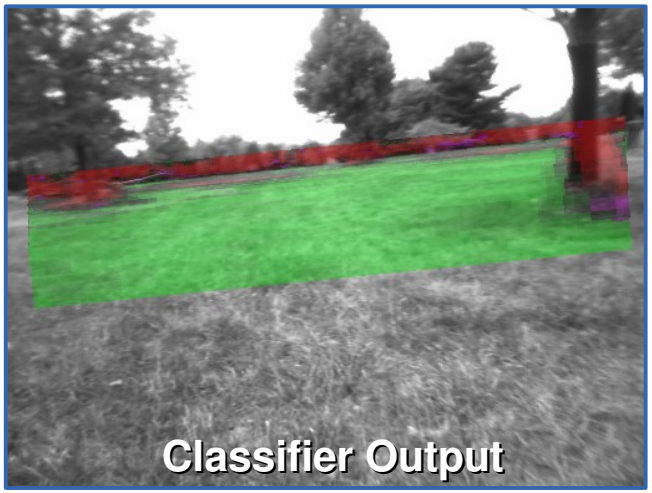


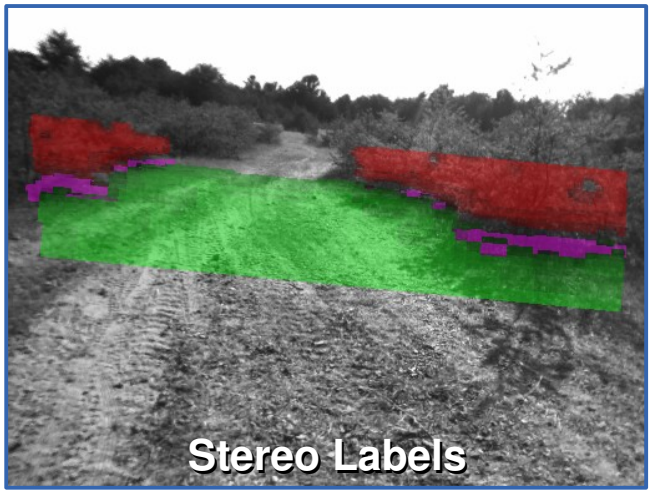
- No post-processing
- Frame-by-frame
- ConvNet runs at 50ms/frame on Virtex-6 FPGA hardware
  - But communicating the features over ethernet limits system performance

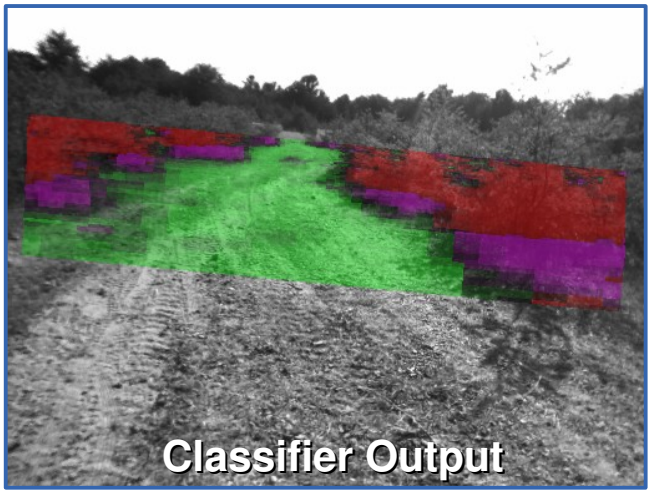
ConvNet for Long Range Adaptive Robot Vision (DARPA LAGR program 2005-2008)

Y LeCun



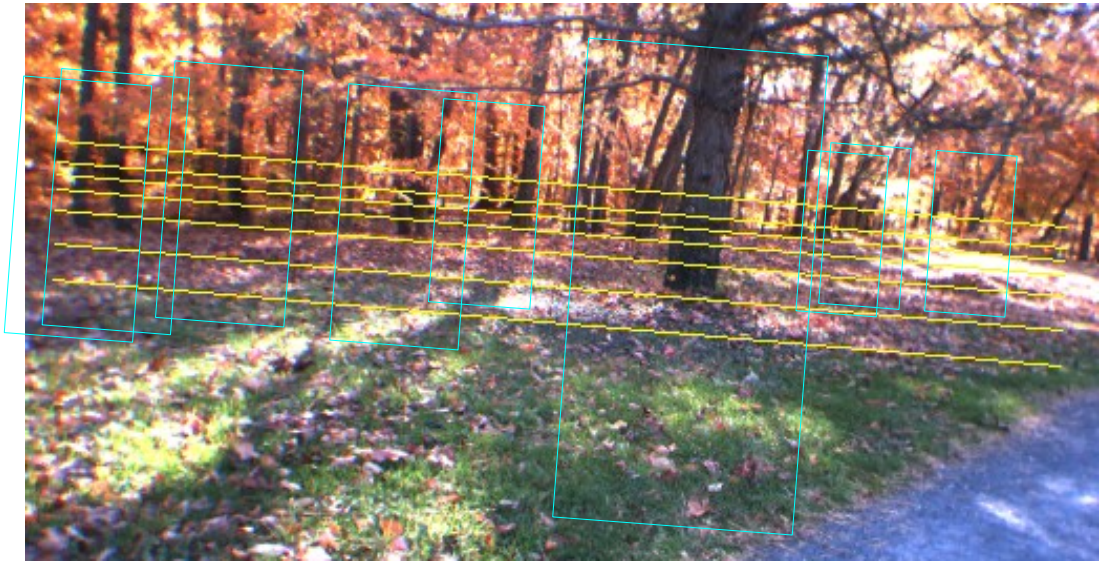






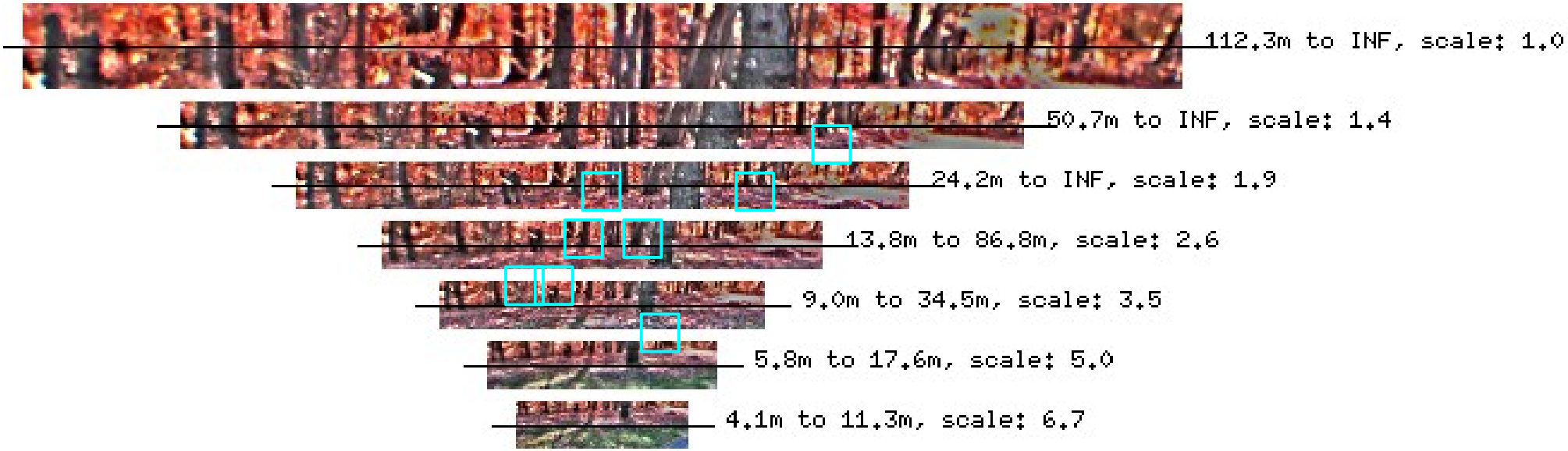
Long Range Vision with a Convolutional Net

Y LeCun



Pre-processing (125 ms)

- Ground plane estimation
- Horizon leveling
- Conversion to YUV + local contrast normalization
- Scale invariant pyramid of distance-normalized image "bands"



Convolutional Net Architecture

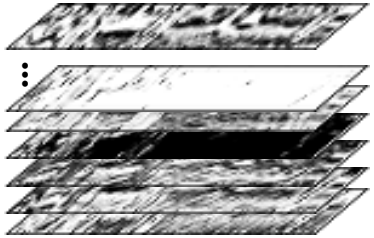
100 features per
3x12x25 input window

100@25x121

CONVOLUTIONS (6x5)

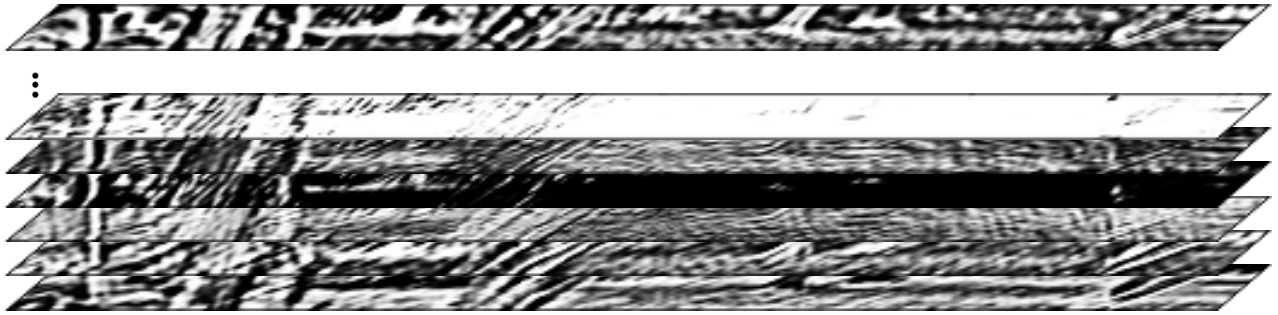
YUV image band
20-36 pixels tall,
36-500 pixels wide

20@30x125



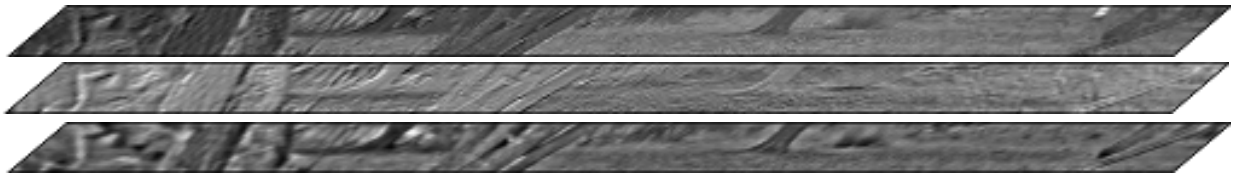
MAX SUBSAMPLING (1x4)

20@30x484



CONVOLUTIONS (7x6)

3@36x484



YUV input

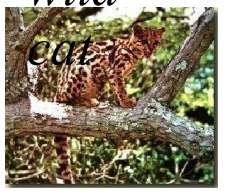
Visual Object Recognition with Convolutional Nets

- In the mid 2000s, ConvNets were getting decent results on object classification
- Dataset: "Caltech101":
  - 101 categories
  - 30 training samples per category
- But the results were slightly worse than more "traditional" computer vision methods, because
  - 1. the datasets were too small
  - 2. the computers were too slow

face

beaver

wild cat

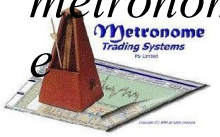


lotus

ant

dollar

metronome



w. chair

minar

cellphon

joshua t.

cougar body

backgroun d

Late 2000s: we could get decent results on object recognition

- But we couldn't beat the state of the art because the datasets were too small
- Caltech101: 101 categories, 30 samples per category.
- But we learned that rectification and max pooling are useful! [Jarrett et al. ICCV 2009]

Single Stage System: $[64.F_{CSG}^{9\times9} - R/N/P^{5\times5}]$ **- log_reg**					
R/N/P	$R_{abs} - N - P_A$	$R_{abs} - P_A$	$N - P_M$	$N - P_A$	P_A
U^+	54.2%	50.0%	44.3%	18.5%	14.5%
R^+	54.8%	47.0%	38.0%	16.3%	14.3%
U	52.2%	43.3%(±1.6)	44.0%	17.2%	13.4%
R	53.3%	31.7%	32.1%	15.3%	12.1%(±2.2)
G	52.3%				
Two Stage System: $[64.F_{CSG}^{9\times9} - R/N/P^{5\times5}] - [256.F_{CSG}^{9\times9} - R/N/P^{4\times4}]$ **- log_reg**					
R/N/P	$R_{abs} - N - P_A$	$R_{abs} - P_A$	$N - P_M$	$N - P_A$	P_A
U^+U^+	65.5%	60.5%	61.0%	34.0%	32.0%
R^+R^+	64.7%	59.5%	60.0%	31.0%	29.7%
UU	63.7%	46.7%	56.0%	23.1%	9.1%
RR	62.9%	33.7%(±1.5)	37.6%(±1.9)	19.6%	8.8%
GT	55.8%	← like HMAX model			
Single Stage: $[64.F_{CSG}^{9\times9} - R/N/P^{5\times5}]$ **- PMK-SVM**					
U	64.0%				
Two Stages: $[64.F_{CSG}^{9\times9} - R/N/P^{5\times5}] - [256.F_{CSG}^{9\times9} - R/N]$ **- PMK-SVM**					
UU	52.8%				

Then., two things happened...

- **The ImageNet dataset [Fei-Fei et al. 2012]**
  - 1.2 million training samples
  - 1000 categories

- **Fast & Programmable General-Purpose GPUs**
  - NVIDIA CUDA
  - Capable of over 1 trillion operations/second

Matchstick

Flute

Backpack

Sea lion

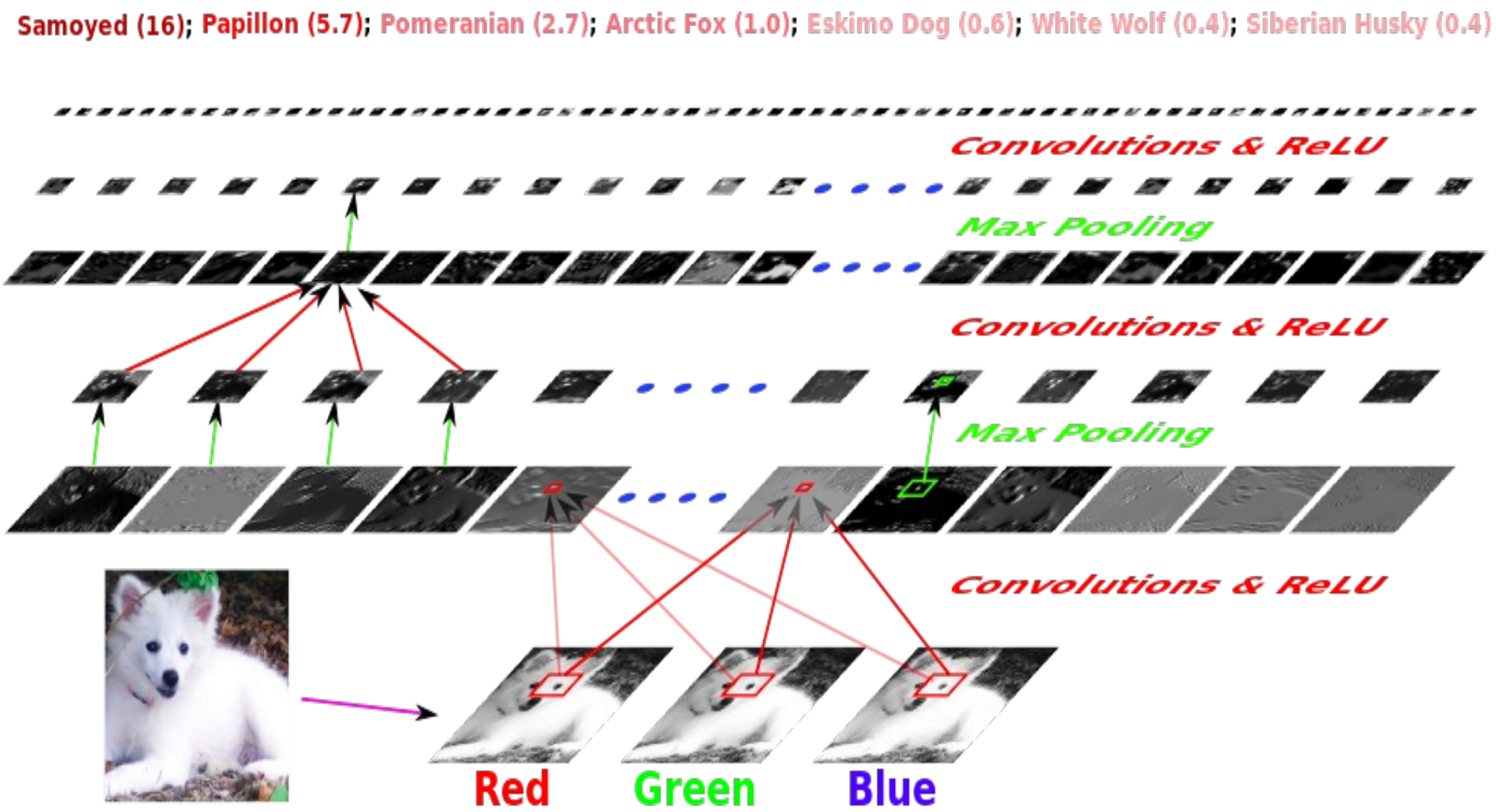
Strawberry

Bathing cap

Racket

Very Deep ConvNet for Object Recognition

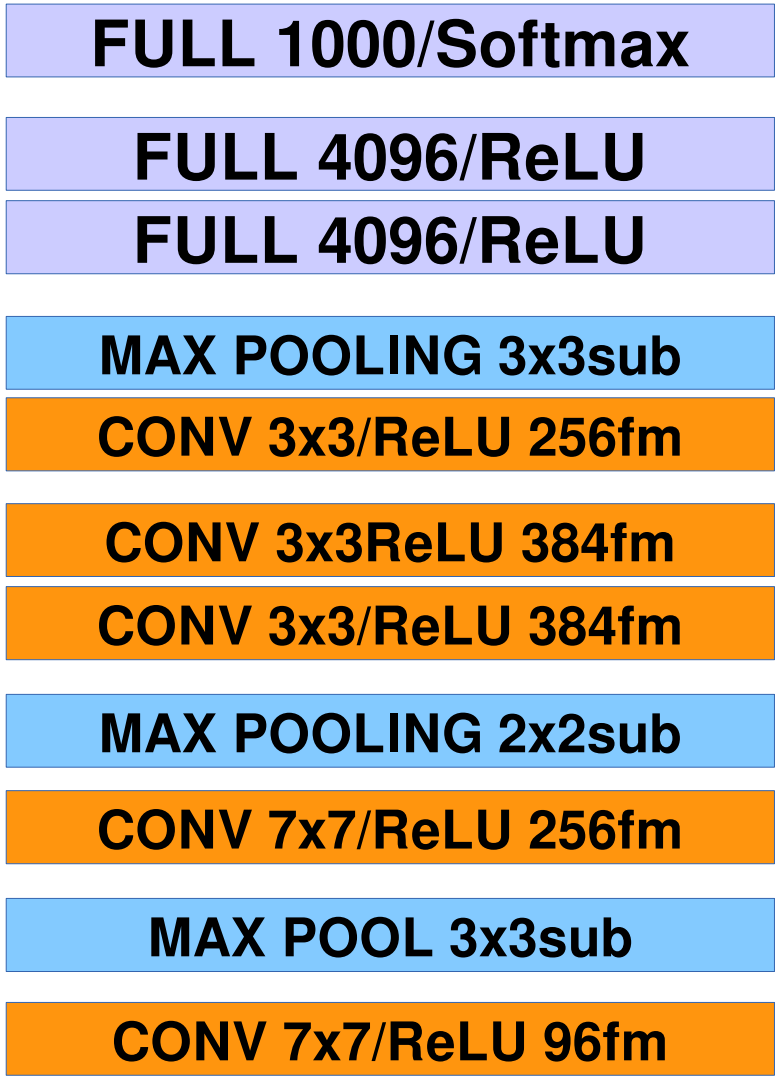
- 1 to 10 billion connections, 10 million to 1 billion parameters, 8 to 20 layers.



Very Deep ConvNets Trained on GPU

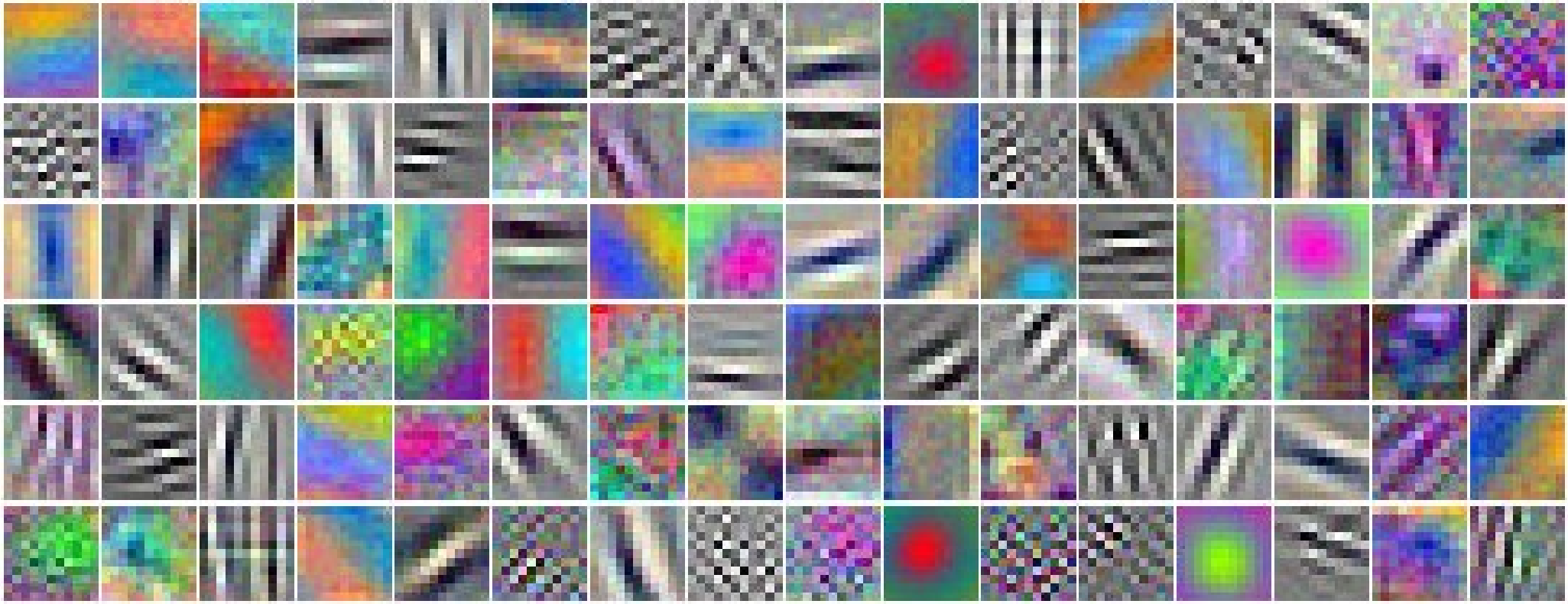
- AlexNet [Krizhevski, Sutskever, Hinton 2012]
  - 15% top-5 error on ImageNet
  - Open source implementations: CudaConvNet, Torch7, Caffe
- OverFeat [Sermanet et al. 2013]
  - 13.8%
  - Torch7
- VGG Net [Simonyan, Zisserman 2014]
  - 7.3%
  - Torch7, Caffe
- GoogLeNet [Szegedy et al. 2014]
  - 6.6%
  - Torch7, Caffe

- http://torch.ch
- https://github.com/torch/torch7/wiki/Cheatsheet



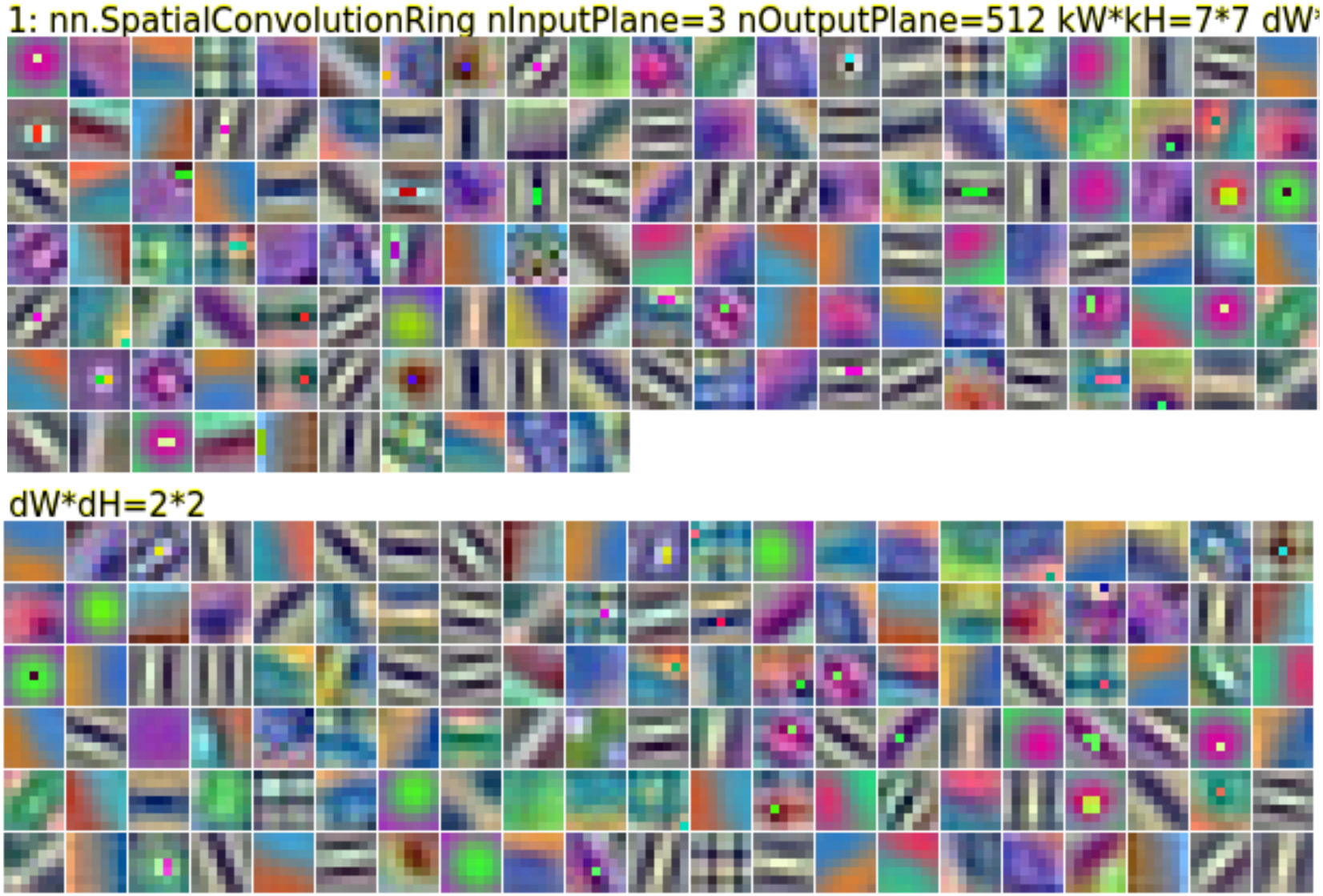
Kernels: Layer 1 (11x11)

- Layer 1: 3x96 kernels, RGB->96 feature maps, 11x11 Kernels, stride 4



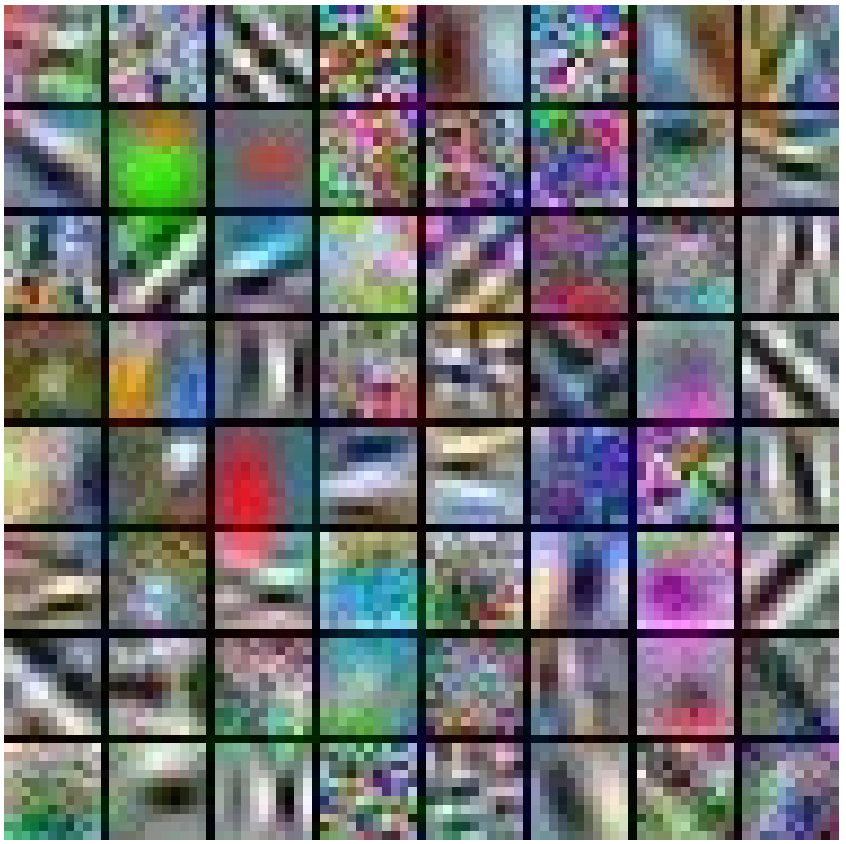
Kernels: Layer 1 (11x11)

Layer 1: 3x512 kernels, 7x7, 2x2 stride.



Learning in Action

- **How the filters in the first layer learn**

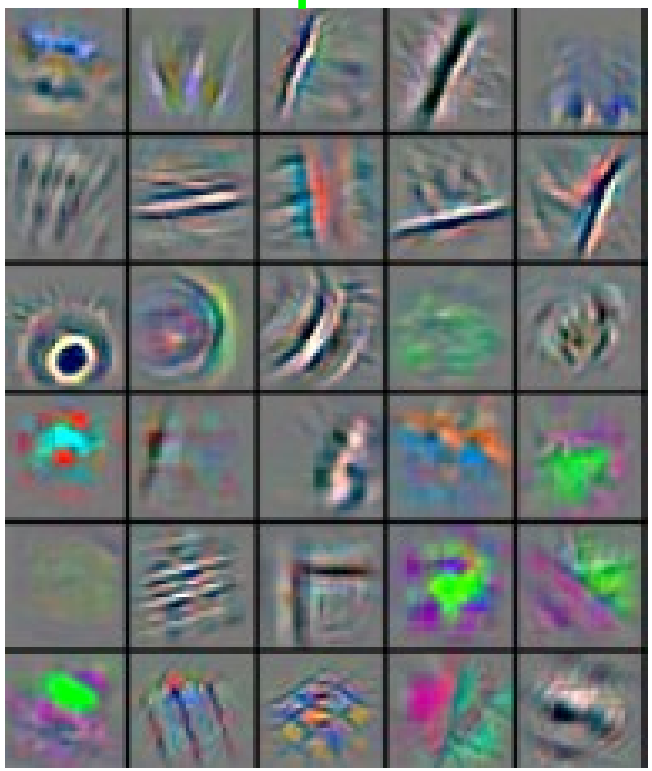


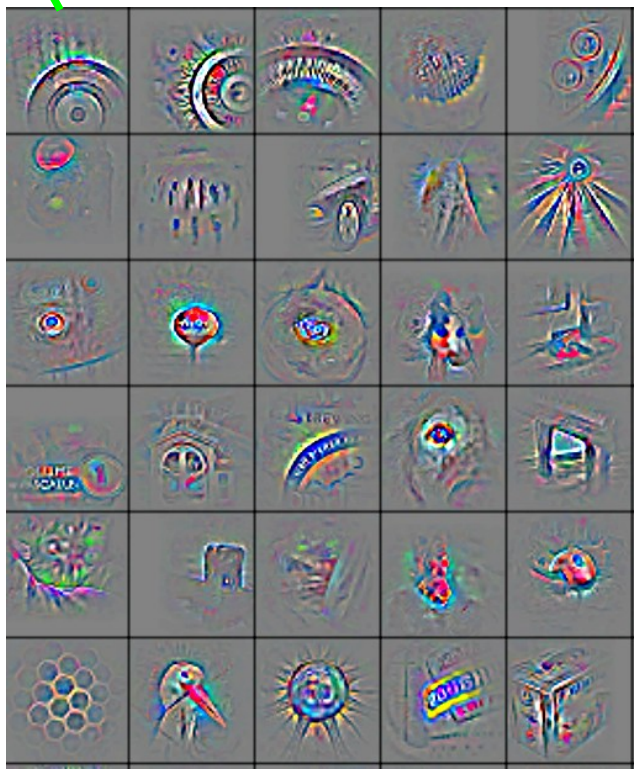
Deep Learning = Learning Hierarchical Representations

Y LeCun

- It's deep if it has more than one stage of non-linear feature transformation

Low-Level Feature → Mid-Level Feature → High-Level Feature → Trainable Classifier





Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

ImageNet: Classification

- Give the name of the dominant object in the image
- Top-5 error rates: if correct class is not in top 5, count as error
  - Red:ConvNet, blue: no ConvNet

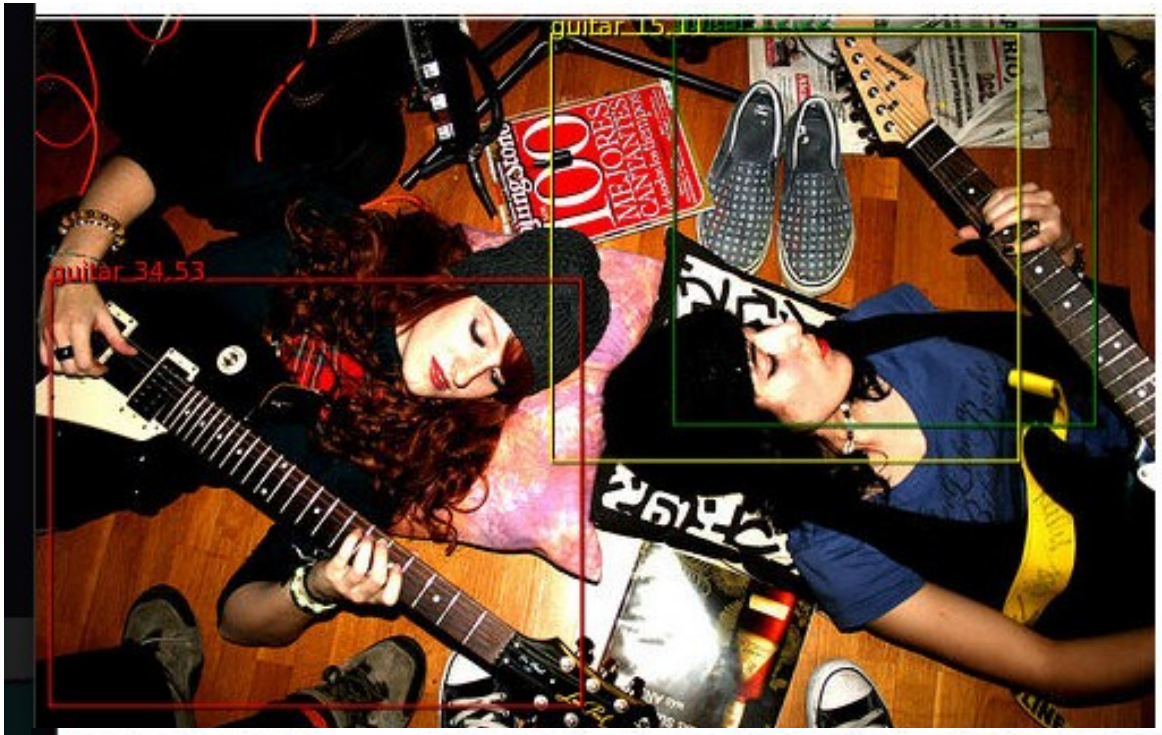
2012 Teams	%error
Supervision (Toronto)	15.3
ISI (Tokyo)	26.1
VGG (Oxford)	26.9
XRCE/INRIA	27.0
UvA (Amsterdam)	29.6
INRIA/LEAR	33.4

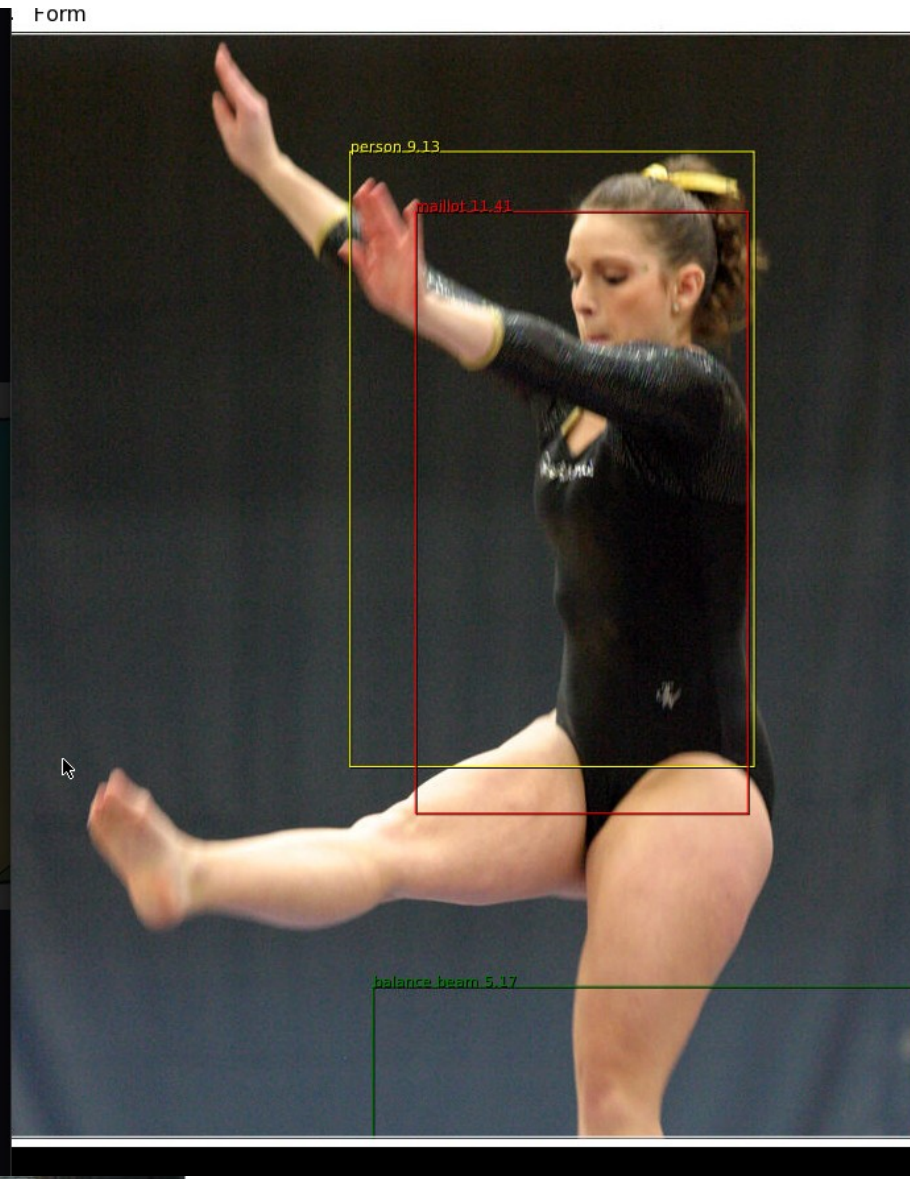
2013 Teams	%error
Clarifai (NYU spinoff)	11.7
NUS (singapore)	12.9
Zeiler-Fergus (NYU)	13.5
A. Howard	13.5
OverFeat (NYU)	14.1
UvA (Amsterdam)	14.2
Adobe	15.2
VGG (Oxford)	15.2
VGG (Oxford)	23.0

2014 Teams	%error
GoogLeNet	6.6
VGG (Oxford)	7.3
MSRA	8.0
A. Howard	8.1
DeeperVision	9.5
NUS-BST	9.7
TTIC-ECP	10.2
XYZ	11.2
UvA	12.1

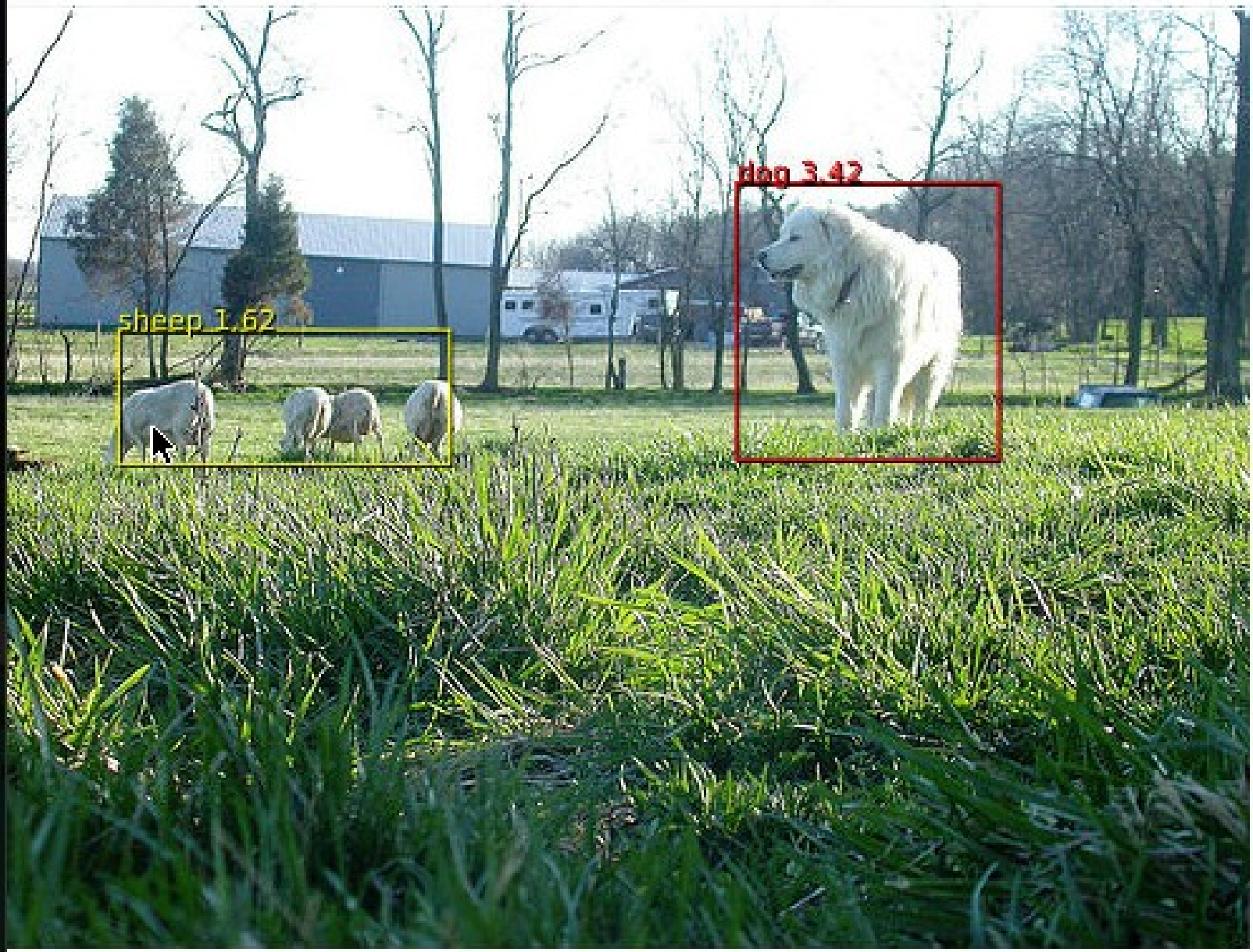
Results: pre-trained on ImageNet1K, fine-tuned on ImageNet Detection

Y LeCun



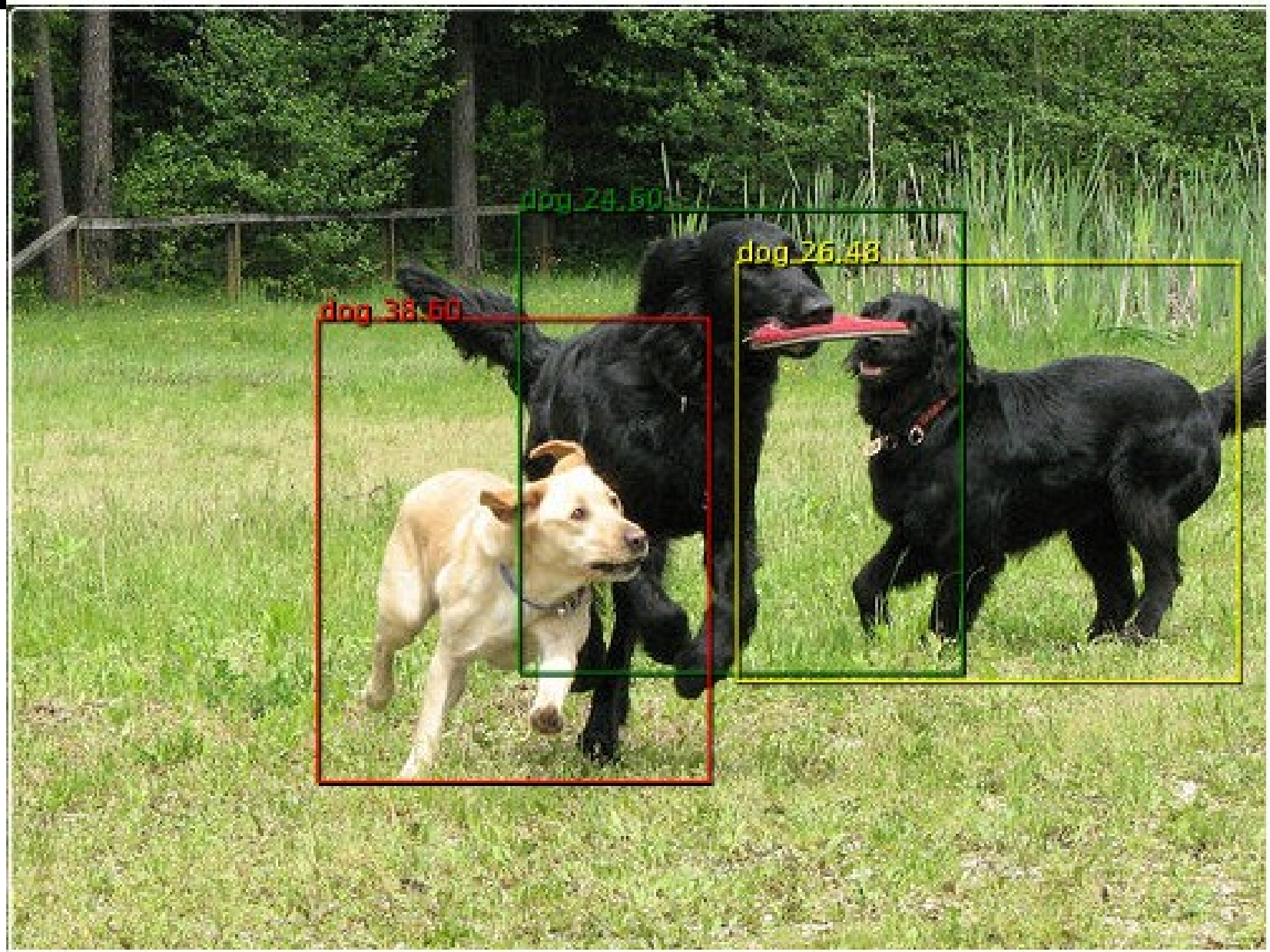


Detection Examples



/home/snwiz/data/imagenet12/original/det/ILSVRC2013_DET_test/ILSVRC2012_test_00090628.JPEG
dog conf 3.419652

Detection Examples

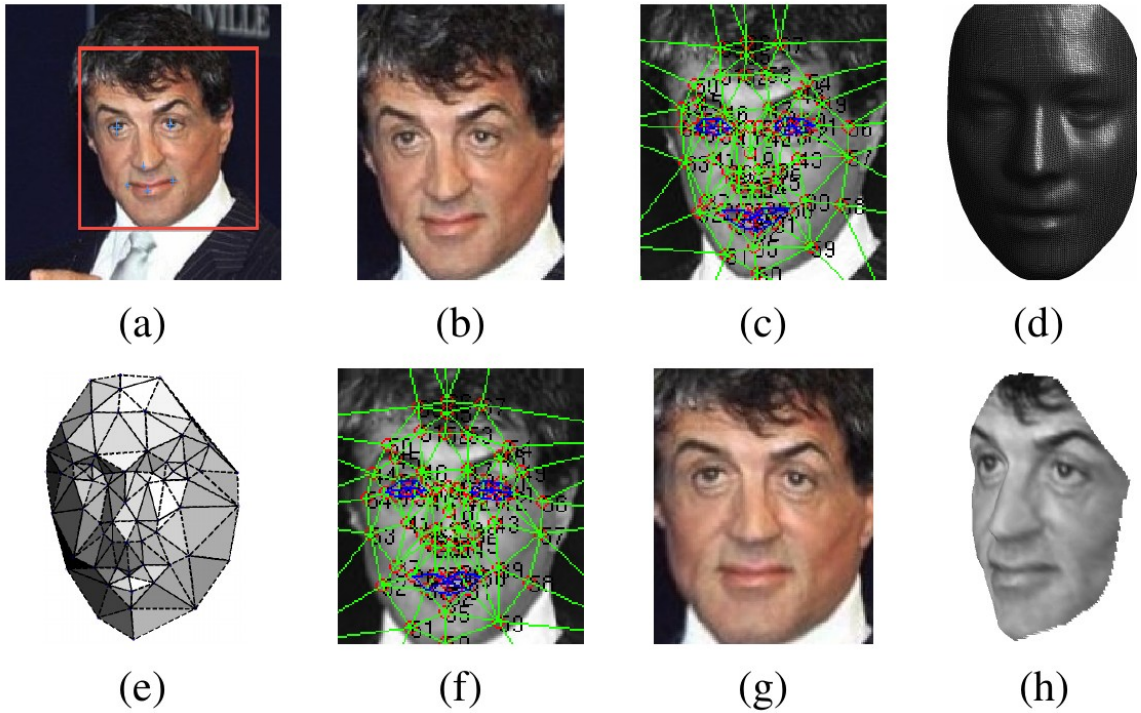


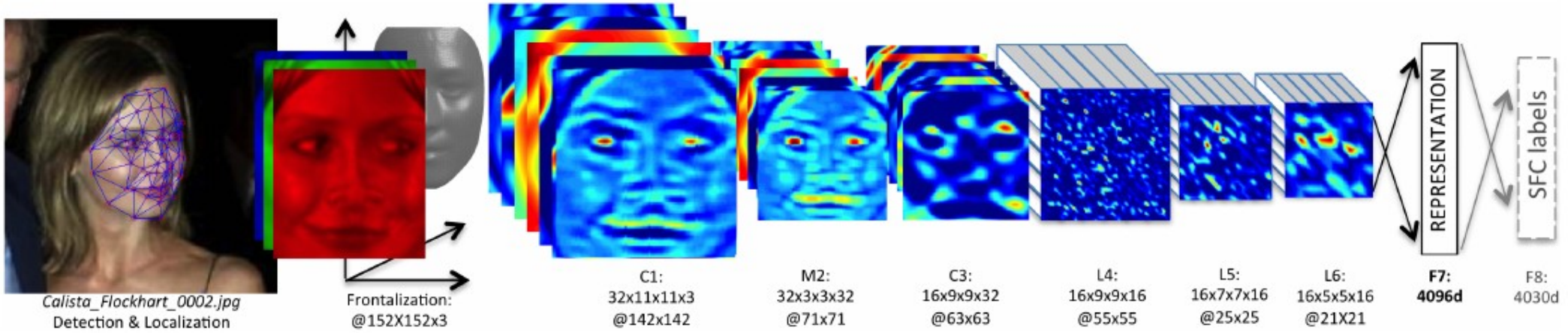
/home/snwiz/data/imagenet12/original/det/ILSVRC2013_DET_test/ILSVRC2012_test_00000172.JPEG
dog conf 38.603936


WE PROVIDI
THIS SERVICE
YOUR
person 9.06
french horn 21.89

Face Recognition:DeepFace (Facebook AI Research)

- [Taigman et al. CVPR 2014]
  - Alignment
  - Convnet
  - Close to human performance on frontal views
  - Can now look for a person among 800 millions in 5 seconds
  - Uses 256-bit "compact binary codes"
  - [Gong et al CVPR 2015]

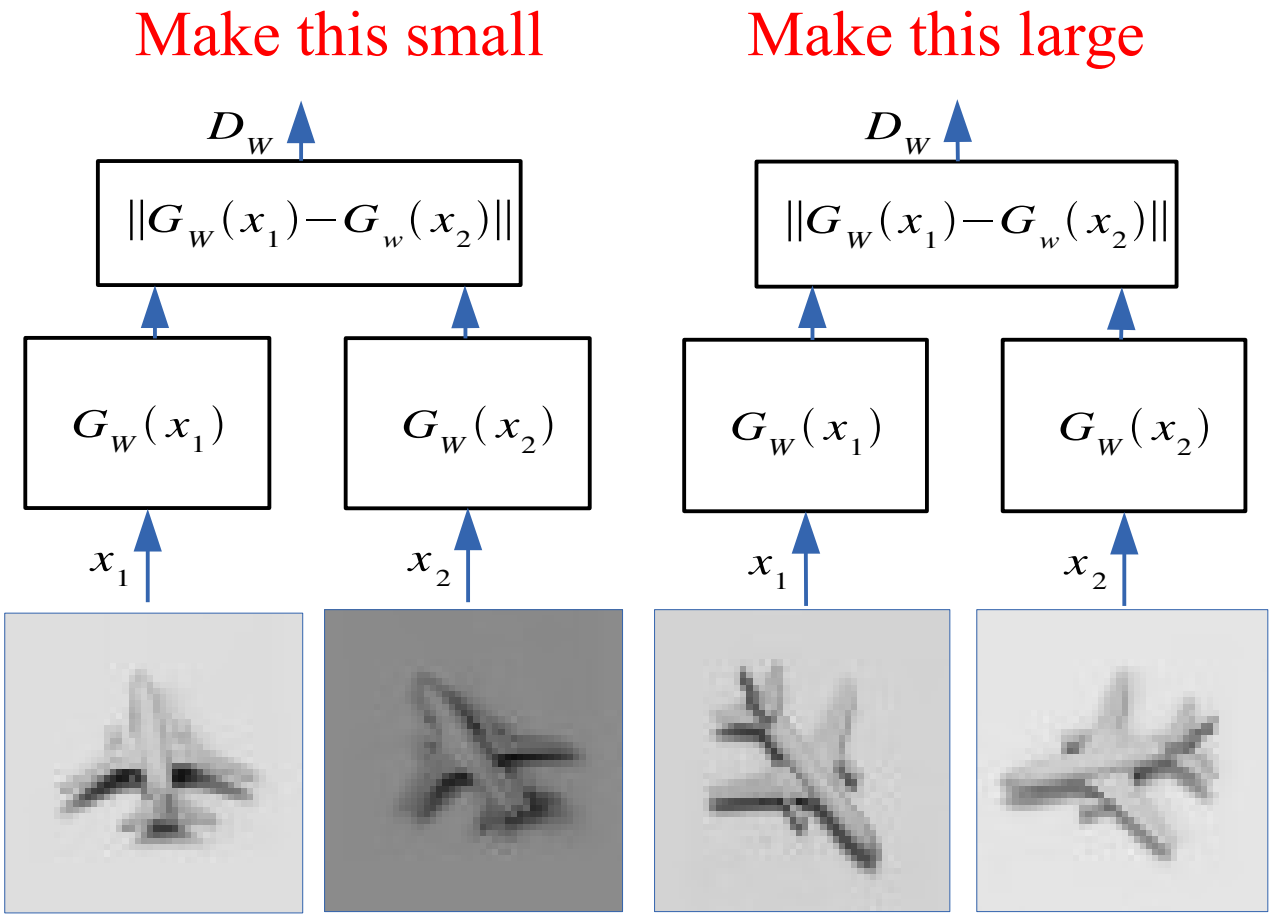




Siamese Architecture and loss function

Loss function:

- Outputs corresponding to input samples that are neighbors in the neigborhood graph should be nearby
- Outputs for input samples that are not neighbors should be far away from each other



Segmenting and Localizing Objects

- [Pinheiro, Collobert, Dollar 2015]
  - ConvNet produces object masks

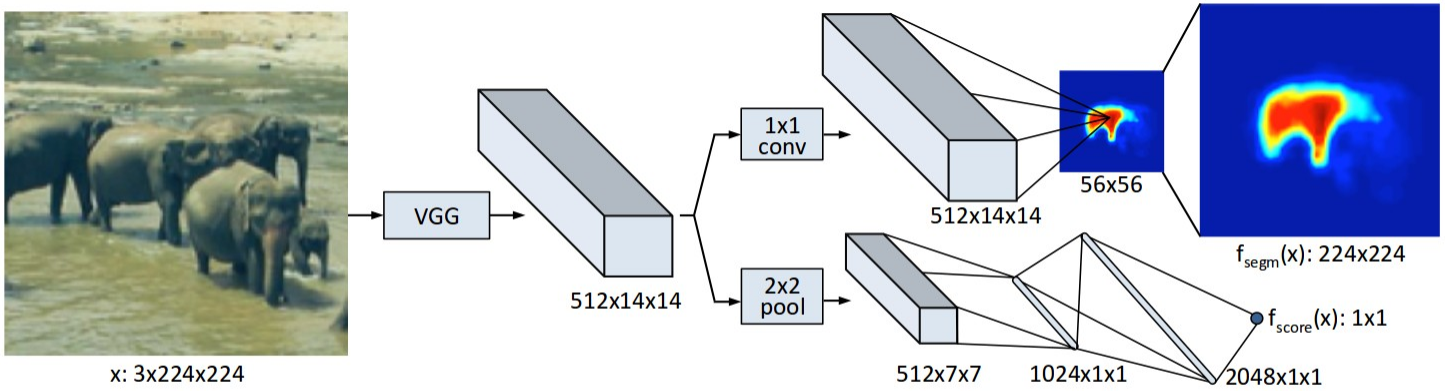


Image captioning: generating a descriptive sentence

Y LeCun

- [Lebret, Pinheiro, Collobert 2015]
- [Kulkarni 11] [Mitchell 12] [Vinyals 14] [Mao 14] [Karpathy 14] [Donahue 14]...

A man riding skis on a snow covered ski slope.

NP: a man, skis, the snow, a person, a woman, a snow covered slope, a slope, a snowboard, a skier, man.
VP: wearing, riding, holding, standing on, skiing down.
PP: on, in, of, with, down.

A man wearing skis on the snow.

A man is doing skateboard tricks on a ramp.

NP: a skateboard, a man, a trick, his skateboard, the air, a skateboarder, a ramp, a skate board, a person, a woman.
VP: doing, riding, is doing, performing, flying through.
PP: on, of, in, at, with.

A man riding a skateboard on a ramp.

The girl with blue hair stands under the umbrella.

NP: a woman, an umbrella, a man, a person, a girl, umbrellas, that, a little girl, a cell phone.
VP: holding, wearing, is holding, holds, carrying.
PP: with, on, of, in, under.

A woman is holding an umbrella.

A slice of pizza sitting on top of a white plate.

NP: a plate, a white plate, a table, pizza, it, a pizza, food, a sandwich, top, a close.
VP: topped with, has, is, sitting on, is on.
PP: of, on, with, in, up.

A table with a plate of pizza on a white plate.

A baseball player swinging a bat on a field.

NP: the ball, a game, a baseball player, a man, a tennis court, a ball, home plate, a baseball game, a batter, a field.
VP: swinging, to hit, playing, holding, is swinging.
PP: on, during, in, at, of.

A baseball player swinging a bat on a baseball field.

A bunch of kites flying in the sky on the beach.

NP: the beach, a beach, a kite, kites, the ocean, the water, the sky, people, a sandy beach, a group.
VP: flying, flies, is flying, flying in, are.
PP: on, of, with, in, at.

People flying kites on the beach.

Body Pose Estimation

Pose Estimation and Attribute Recovery with ConvNets

Y LeCun

Pose-Aligned Network for Deep Attribute Modeling

[Zhang et al. CVPR 2014] (Facebook AI Research)

(a) Highest scoring results for people wearing glasses.

(b) Highest scoring results for people wearing a hat.

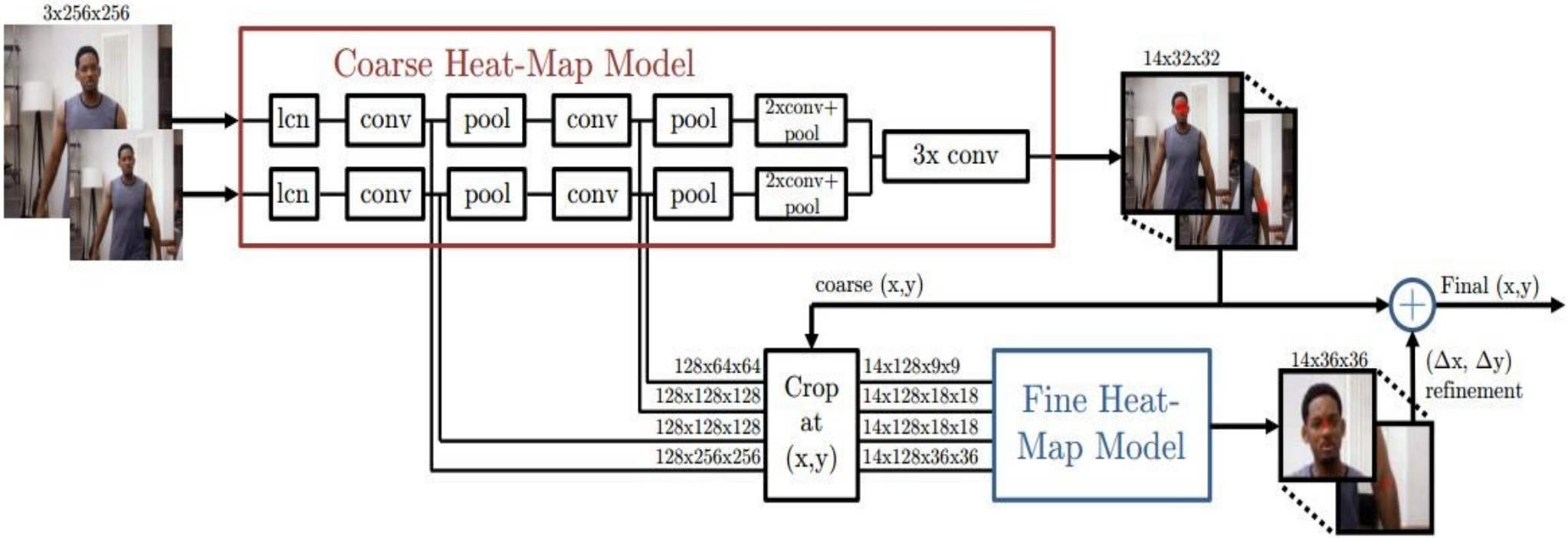
Real-time hand pose recovery

[Tompson et al. Trans. on Graphics 14]

Body pose estimation [Tompson et al. ICLR, 2014]

Person Detection and Pose Estimation

Tompson, Goroshin, Jain, LeCun, Bregler arXiv:1411.4280 (2014)



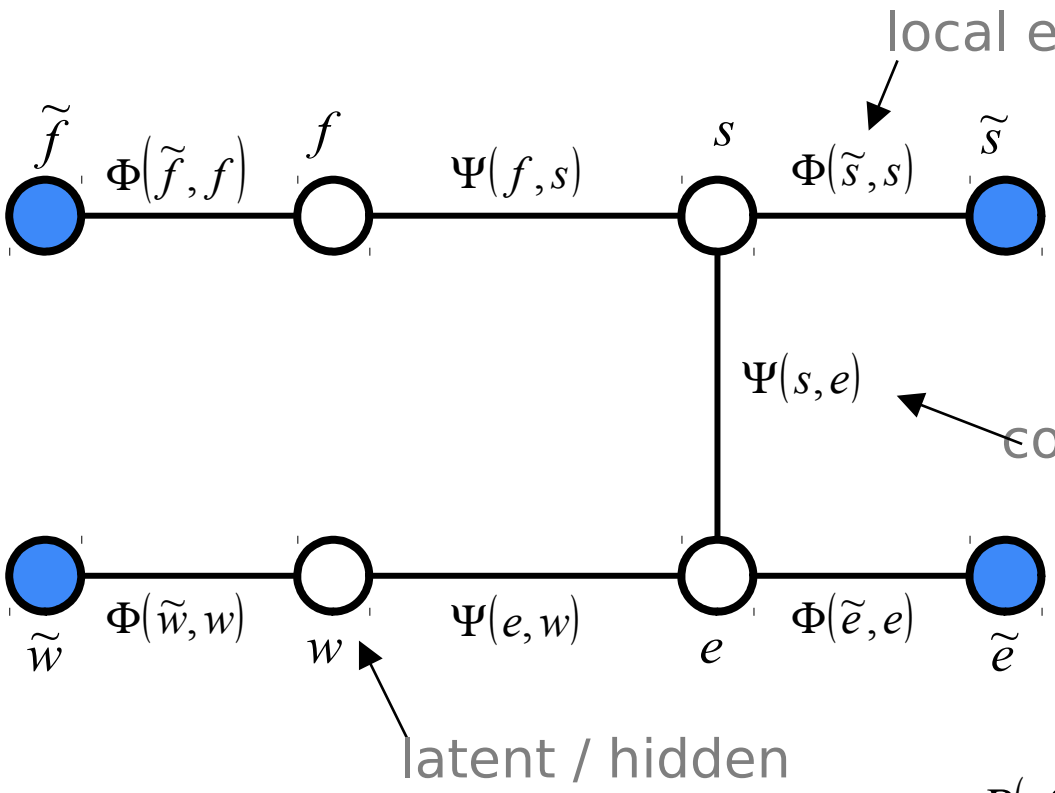
Person Detection and Pose Estimation

Tompson, Goroshin, Jain, LeCun, Bregler arXiv:1411.4280 (2014)

SPATIAL MODEL

Start with a tree graphical model

MRF over spatial locations



Joint Distribution:

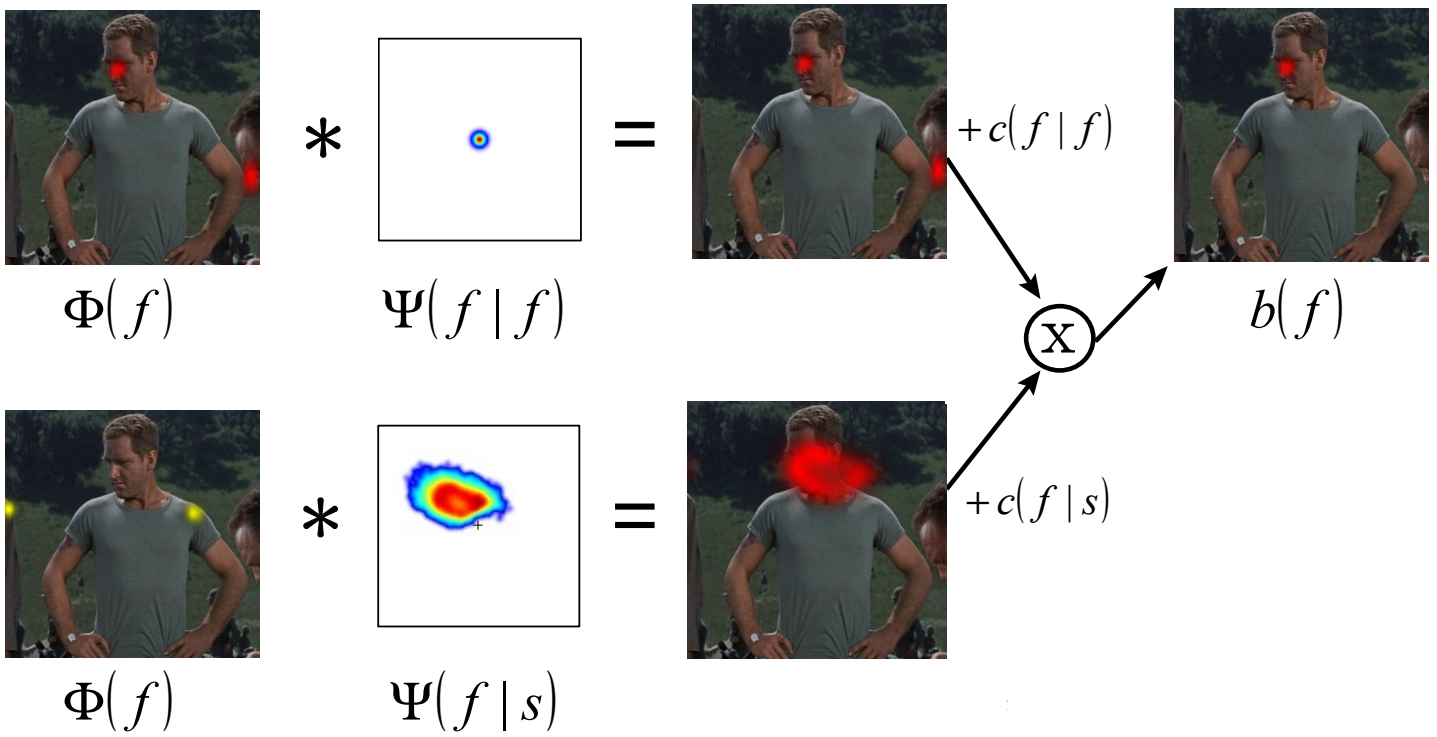
$$P(f,s,e,w)=\frac{1}{Z}\prod_{i,j}\Psi(x_i,x_j)\prod_{i}\Phi(x_i,\tilde{x}_i)$$

SPATIAL MODEL

Start with a tree graphical model

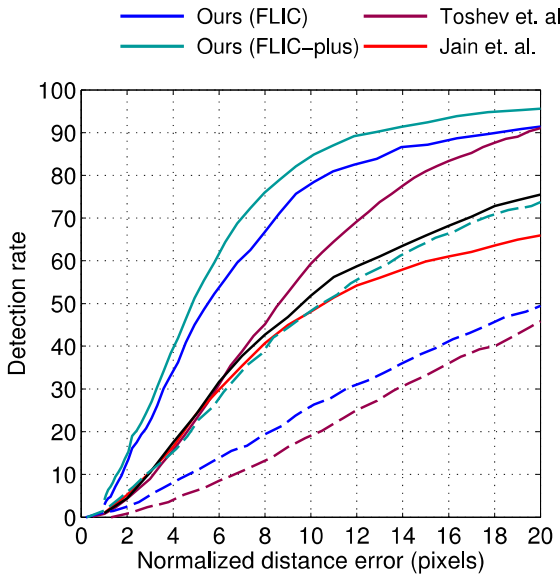
... And approximate it

$$b(f) = \Phi(f) \prod_i \left(\Phi(x_i) * \Psi(f \mid x_i) + c(f \mid x_i) \right)$$

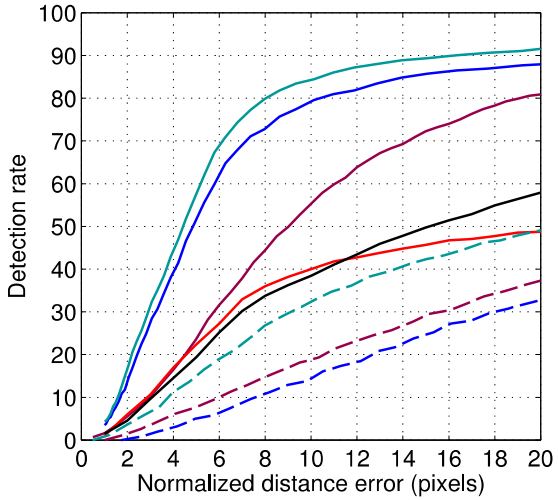


SPATIAL MODEL: RESULTS

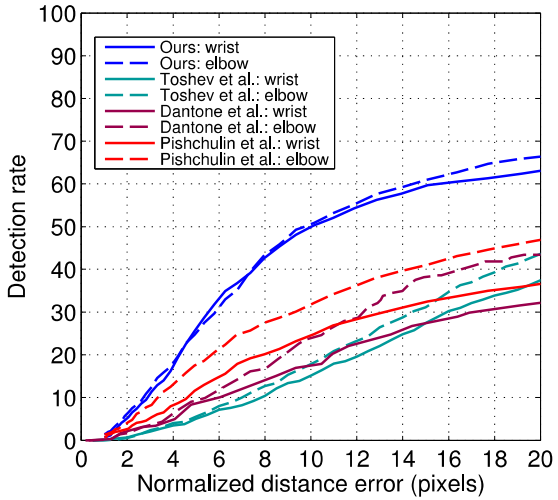
FLIC[1] Elbow



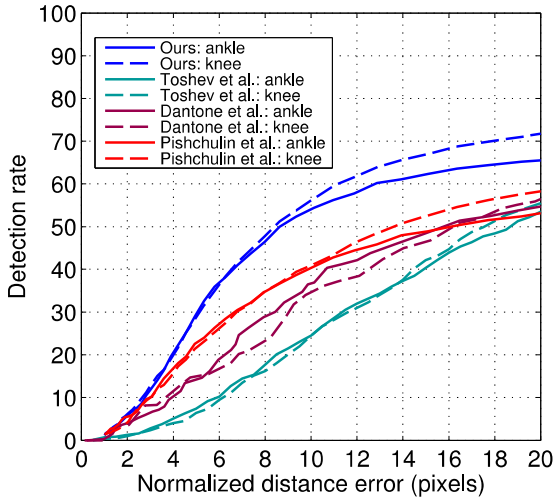
FLIC[1] Wrist



LSP[2] Arms



LSP[1] Legs



(1) B. Sapp and B. Taskar. MODEC: Multimodel decomposition models for human pose estimation. CVPR'13
(2) S. Johnson and M. Everingham. Learning Effective Human Pose Estimation for Inaccurate Annotation. CVPR'11


Video Classification

Learning Video Features with C3D

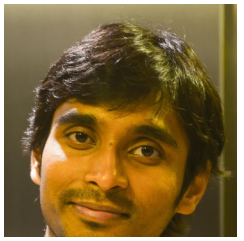
- C3D Architecture
  - 8 convolution, 5 pool, 2 fully-connected layers
  - 3x3x3 convolution kernels
  - 2x2x2 pooling kernels
- Dataset: Sports-1M [Karpathy et al. CVPR'14]
  - 1.1M videos of 487 different sport categories
  - Train/test splits are provided


Du Tran (1,2)


Lubomir Bourdev (2)


Rob Fergus (2,3)


Lorenzo Torresani (1)


Manohar Paluri (2)

(1) Dartmouth College, (2) Facebook AI Research, (3) New York University

Sport Classification Results

Method	Number of Nets	Clip hit@1	Video hit@1	Video hit@5
Deep Video's Single-Frame + Multires [19]	3 nets	42.4	60.0	78.5
Deep Video's Slow Fusion [19]	1 net	41.9	60.9	80.2
C3D (trained from scratch)	1 net	44.9	60.0	84.4
C3D (fine-tuned from I380K pre-trained model)	1 net	**46.1**	**61.1**	**85.2**

Video Classification

- **Using a spatio-temporal ConvNet**

Sport Classification Results

Method	Number of Nets	Clip hit@1	Video hit@1	Video hit@5
Deep Video's Single-Frame + Multires [19]	3 nets	42.4	60.0	78.5
Deep Video's Slow Fusion [19]	1 net	41.9	60.9	80.2
C3D (trained from scratch)	1 net	44.9	60.0	84.4
C3D (fine-tuned from I380K pre-trained model)	1 net	**46.1**	**61.1**	**85.2**

Video Classification

- **Using a spatio-temporal ConvNet**

Video Classification

- **Spatio-temporal ConvNet**

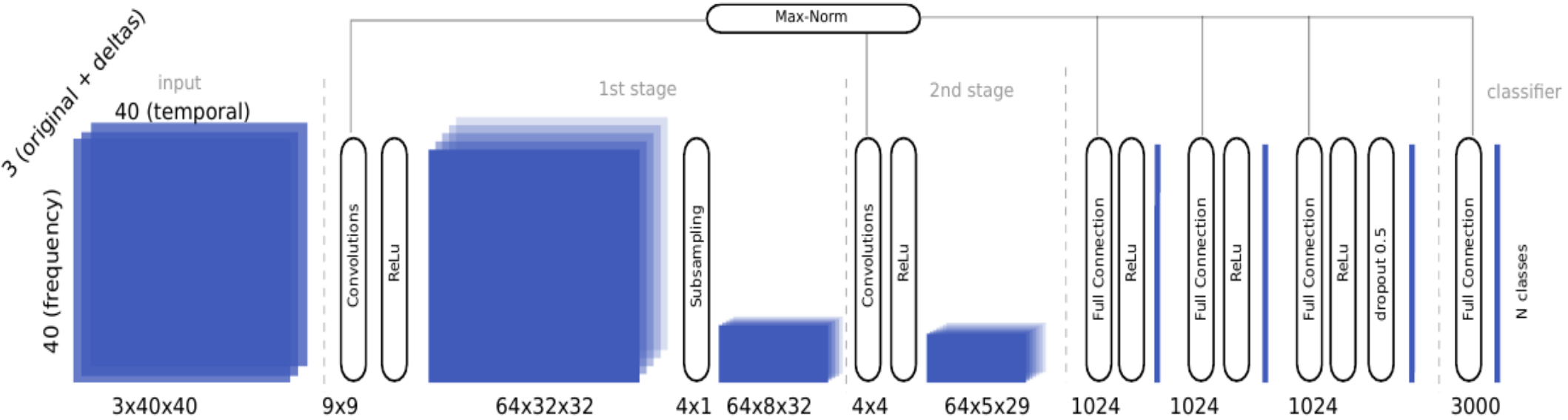
ConvNets are Used Everywhere

- **By Facebook, Google, Microsoft, IBM, Yahoo, Twitter, Baidu, Yandex and a (quickly growing) number of startups**
- **Image recognition, video classification, similarity search, captioning, speech recognition…**
- **These companies collectively process billions of photos per day.**
  - Huge infrastructure
- **Networks for image tagging have typically**
  - 1e9 to 1e11 connections (multiply-accumulate operations per image)
  - 1e7 to 1e9 trainable parameters
  - 10 to 20 layers
  - Trained on 1e7 or more samples.

- **These models seem ridiculously over-parameterized from the statistical point of view. Yet they work really well.**

Speech Recognition with Convolutional Nets (NYU/IBM)

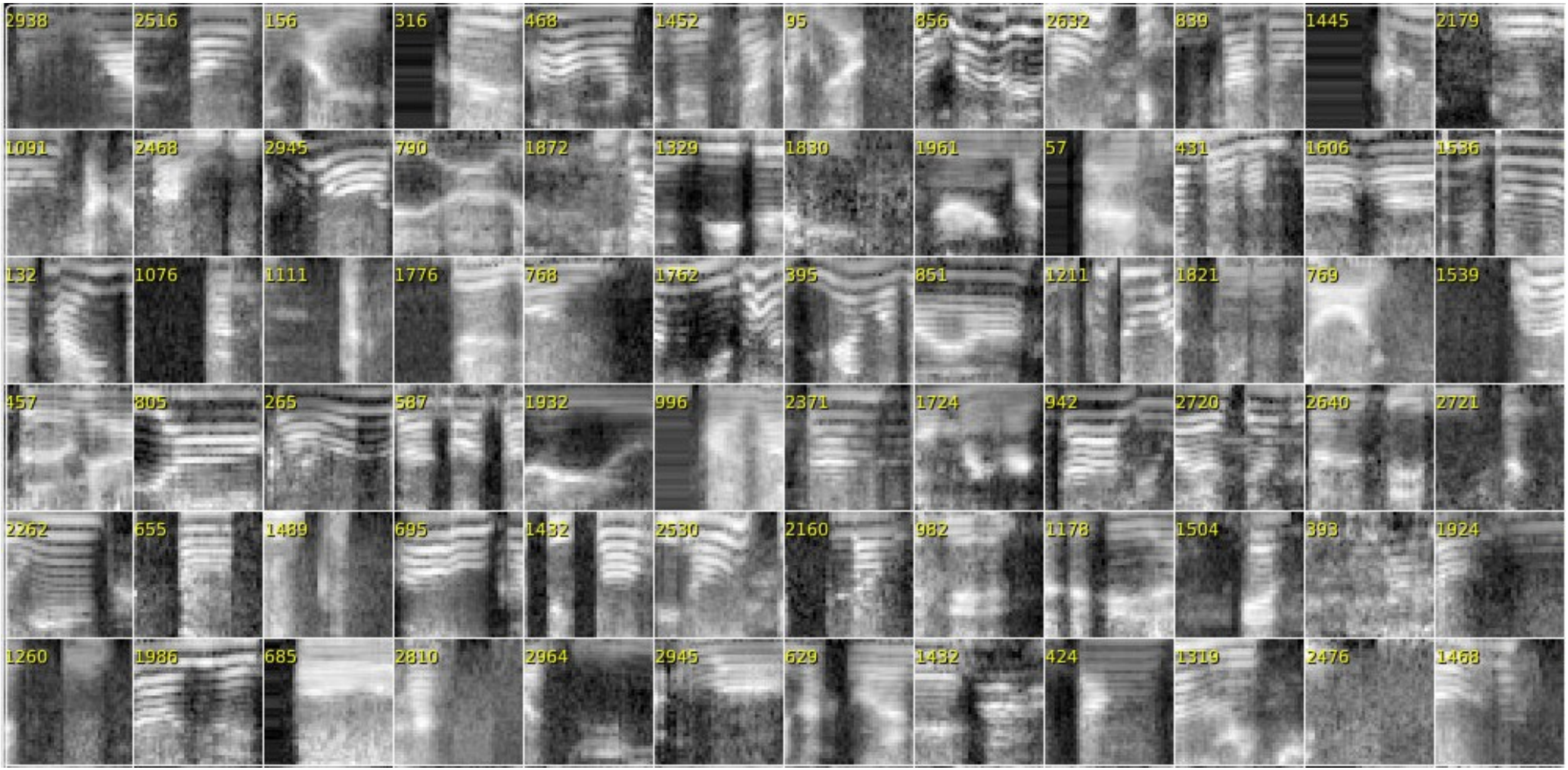
Y LeCun



- Acoustic Model: ConvNet with 7 layers. 54.4 million parameters.
- Classifies acoustic signal into 3000 context-dependent subphones categories
- ReLU units + dropout for last layers
- Trained on GPU. 4 days of training

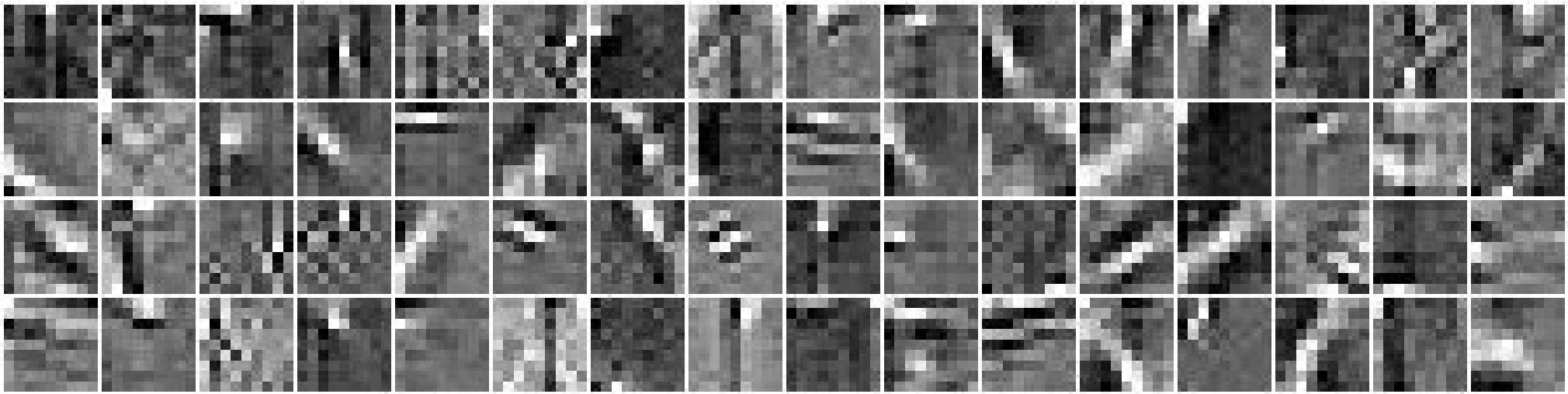
Speech Recognition with Convolutional Nets (NYU/IBM)

- Training samples.
  - 40 MEL-frequency Cepstral Coefficients
  - Window: 40 frames, 10ms each



Speech Recognition with Convolutional Nets (NYU/IBM)

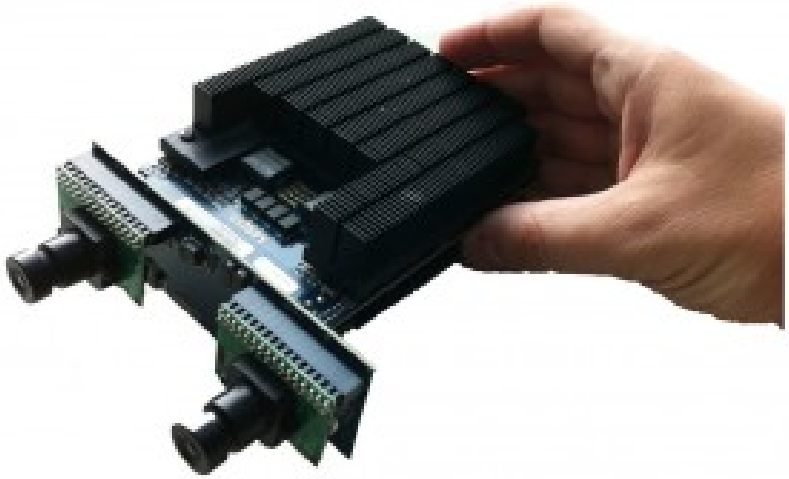
- Convolution Kernels at Layer 1:
  - 64 kernels of size 9x9




ConvNet
Hardware

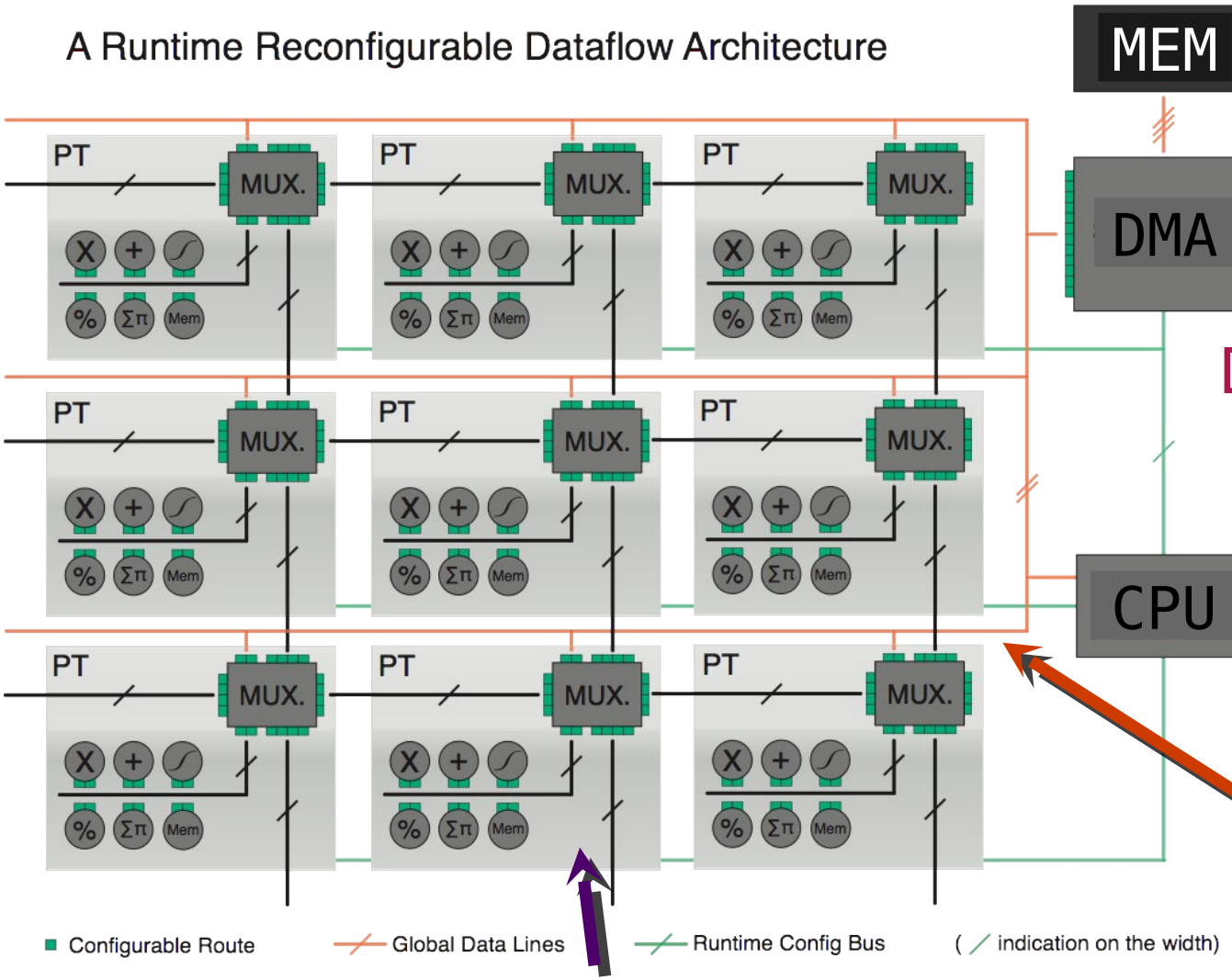
NeuFlow architecture (NYU + Purdue)

- Collaboration NYU-Purdue: Eugenio Culurciello's e-Lab.
- Running on Picocomputing 8x10cm high-performance FPGA board
  - Virtex 6 LX240T: 680 MAC units, 20 neuflow tiles
- Full scene labeling at 20 frames/sec (50ms/frame) at 320x240



board with Virtex-6

NewFlow: Architecture



MEM

DMA

Multi-port memory controller (DMA)

[x12 on a V6 LX240T]

CPU

RISC CPU, to reconfigure tiles and data paths, at runtime

global network-on-chip to allow fast reconfiguration

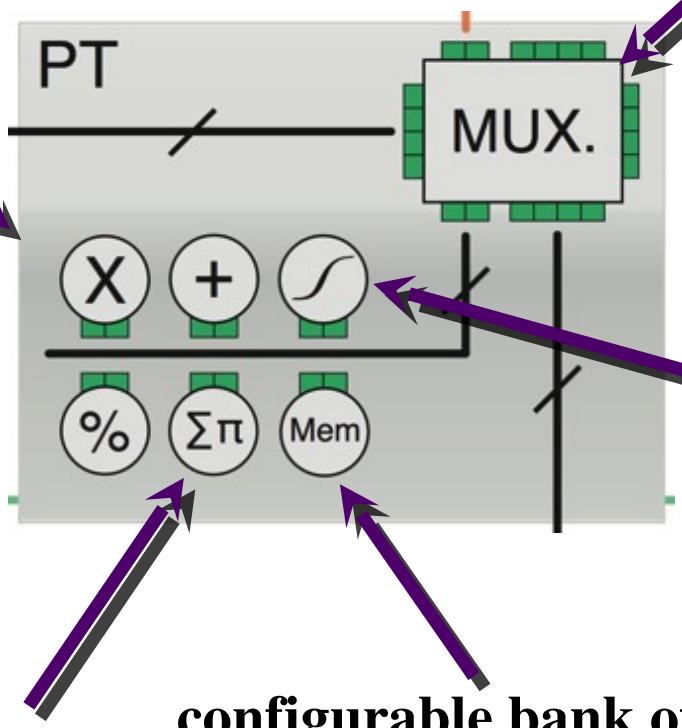
grid of passive processing tiles (PTs)

[x20 on a Virtex6 LX240T]

NewFlow: Processing Tile Architecture

Term-by:-term streaming operators (MUL,DIV,ADD,SUB,MAX)

[x8,2 per tile]



configurable router, to stream data in and out of the tile, to neighbors or DMA ports

[x20]

configurable piece-wise linear or quadratic mapper

[x4]

full 1/2D parallel convolver with 100 MAC units

[x4]

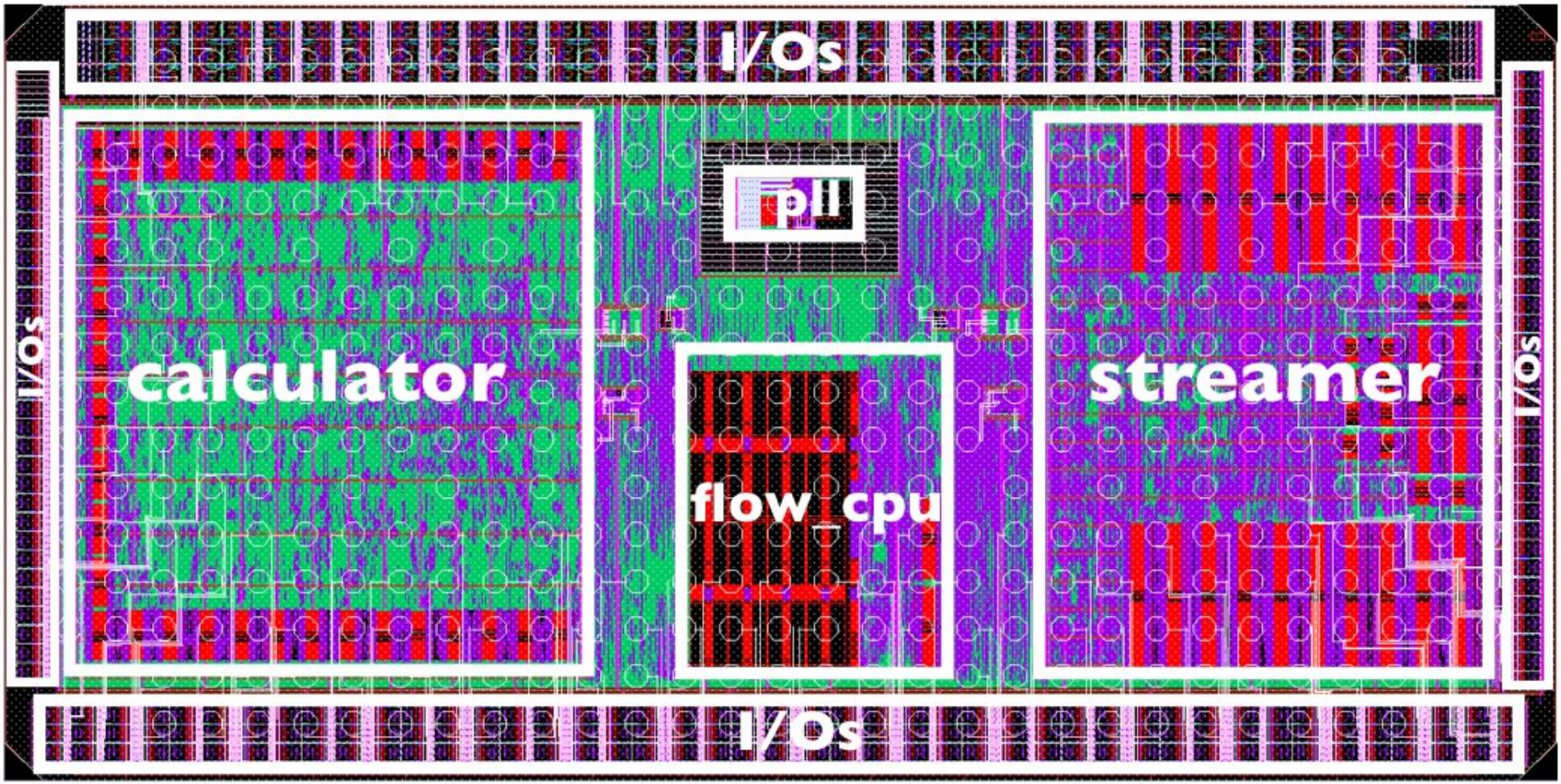
configurable bank of FIFOs , for stream buffering, up to 10kB per PT

[x8]

[Virtex6 LX240T]

NewFlow ASIC: 2.5x5 mm, 45nm, 0.6Watts, >300GOPS

- Collaboration Purdue-NYU: Eugenio Culurciello's e-Lab
- Suitable for vision-enabled embedded and mobile devices
- (but the fabrication was botched...)



[Pham, Jelaca, Farabet, Martini, LeCun, Culurciello 2012]

ConvNet Hardware Today

- **Training is all done on GPUs (on NVIDIA GPUs)**
  - 4 or 8 GPU cards per node. Many nodes per rack. Many racks.
  - 5 to 10 Tflops per card (2015)
  - Training needs performance, programmability, flexibility, accuracy.
  - Power consumption and space is not that important.
- **Many large hardware companies are developing ConvNet accelerators**
  - NVIDIA: evolving from GPU. Embedded applications
    - NVIDIA is investing a lot into deep learning.
  - Intel: ConvNet Fixed Function core. 10X over standard core
  - Movidius: embedded applications
  - Mobileye: ConvNet chip for automotive
  - Orcam: low-power ConvNet chip for the visually impaired
  - Qualcomm, Samsung: ConvNet IP for mobile devices
- **Lots of startups are getting into the field**
  - Teradeep, derived from NeuFlow
  - Nervana, Many others...

Convolutional Nets are Widely Deployed

- **Lots of applications at Facebook, Google, Microsoft, Baidu, Twitter, Yahoo!, IBM…**
  - Image recognition for photo collection search
  - Image/Video Content filtering: spam, nudity, violence.
  - Search, Newsfeed ranking
- **People upload 600 million photos on Facebook every day**
  - (2 billion photos per day if we count Instagram, Messenger and Whatsapp)
- **Each photo on Facebook goes through two ConvNets within 2 seconds**
  - One for image recognition/tagging
  - One for face recognition (not activated in Europe).
- **Soon ConvNets will be everywhere:**
  - self-driving cars, medical image analysis, smart cameras, robots, toys…..


Natural Language
Understanding

Representing Meaning (or Thoughts) with Vectors

Question answering:

- compare vector of question with vectors of answers

Dialog systems:

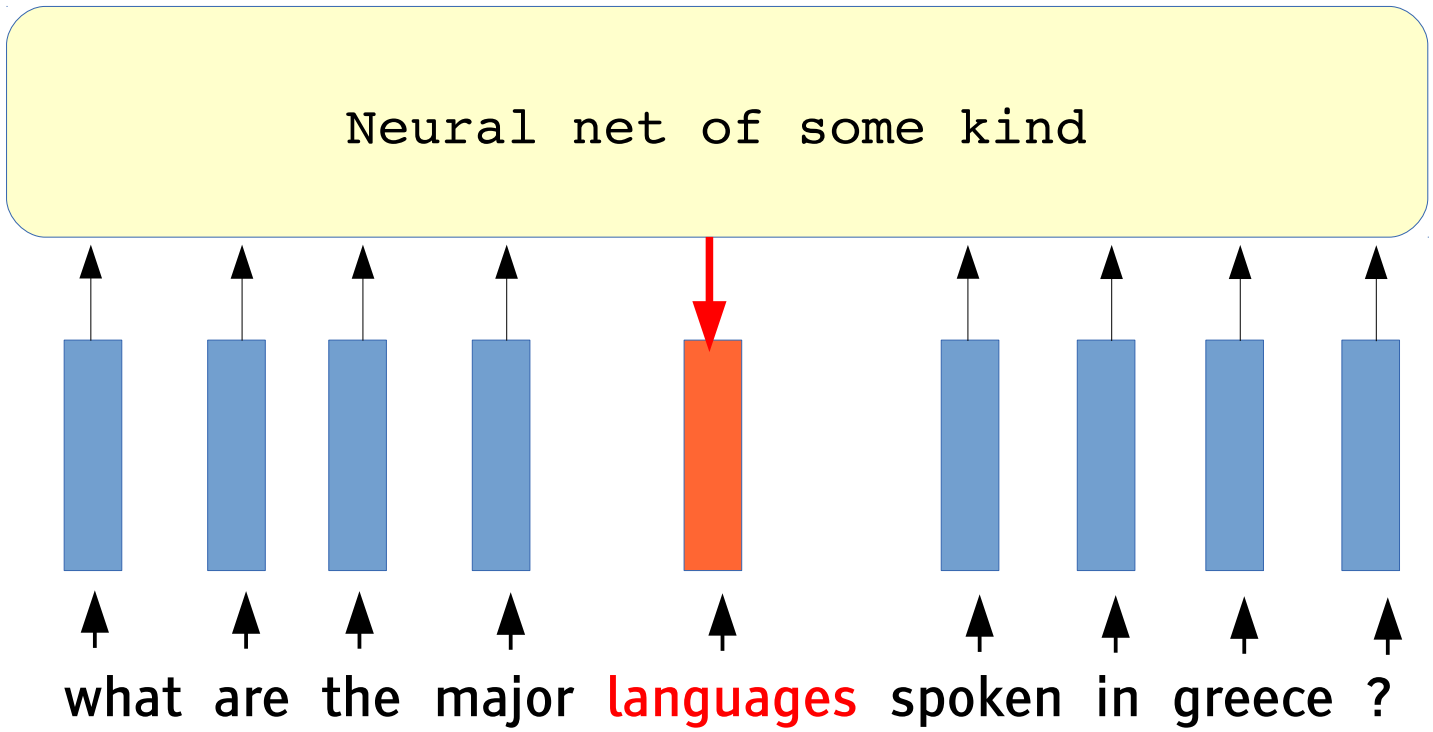
- Turn previous sentences in the dialog into a vector sequence
- Predict the vector of the answer
- Generate the corresponding text

Language Translation:

- Turn the sentence in French into a “meaning vector”
- Generate a sentence in English from the meaning vector.

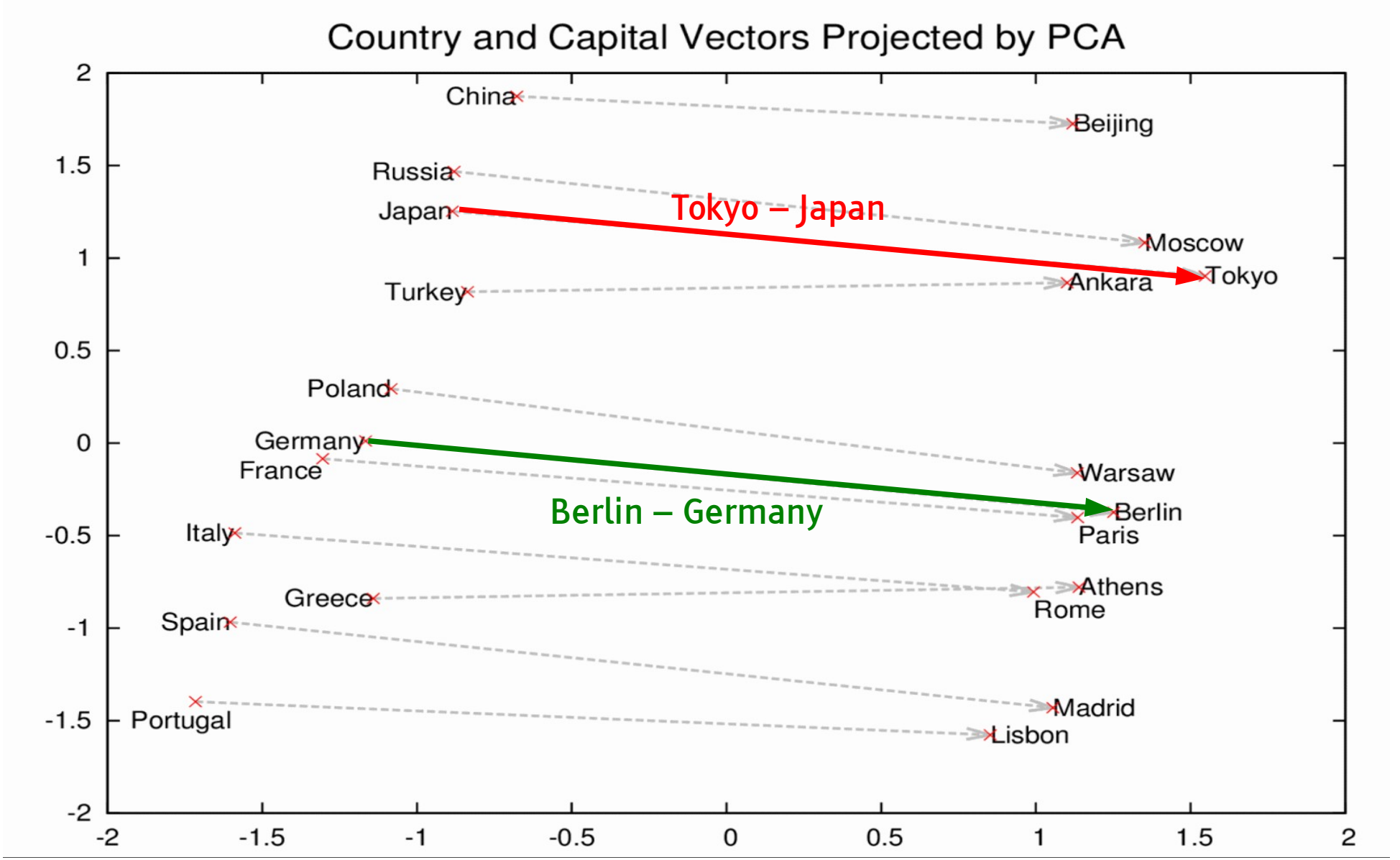
What about Language? Word Embedding

- Word Embedding in continuous vector spaces
  - [Bengio 2003][Collobert & Weston 2010]
  - Word2Vec [Mikolov 2011]
  - Predict a word from previous words and/or following words



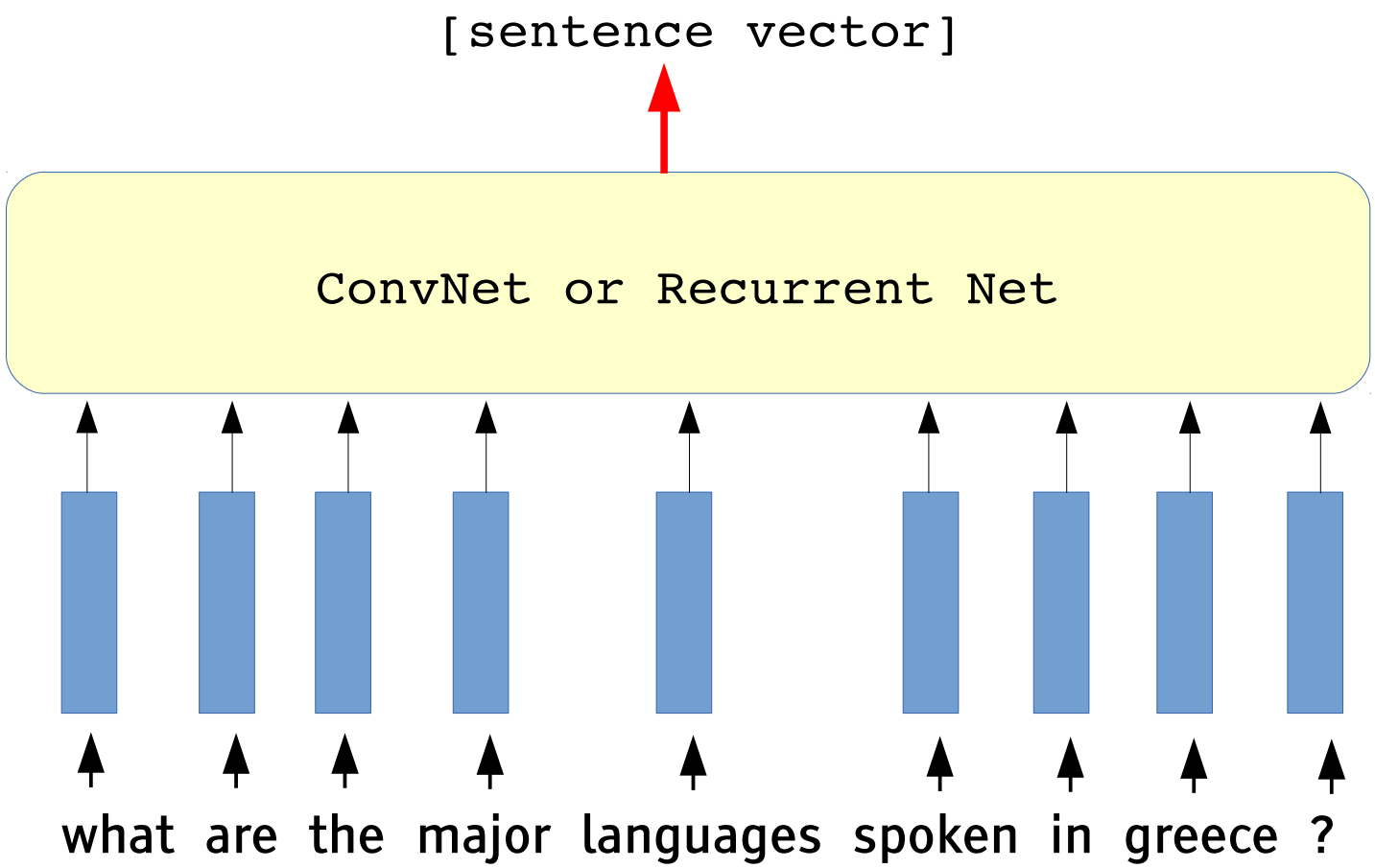
Compositional Semantic Property

Tokyo – Japan = Berlin – Germany Tokyo - Japan + Germany = Berlin

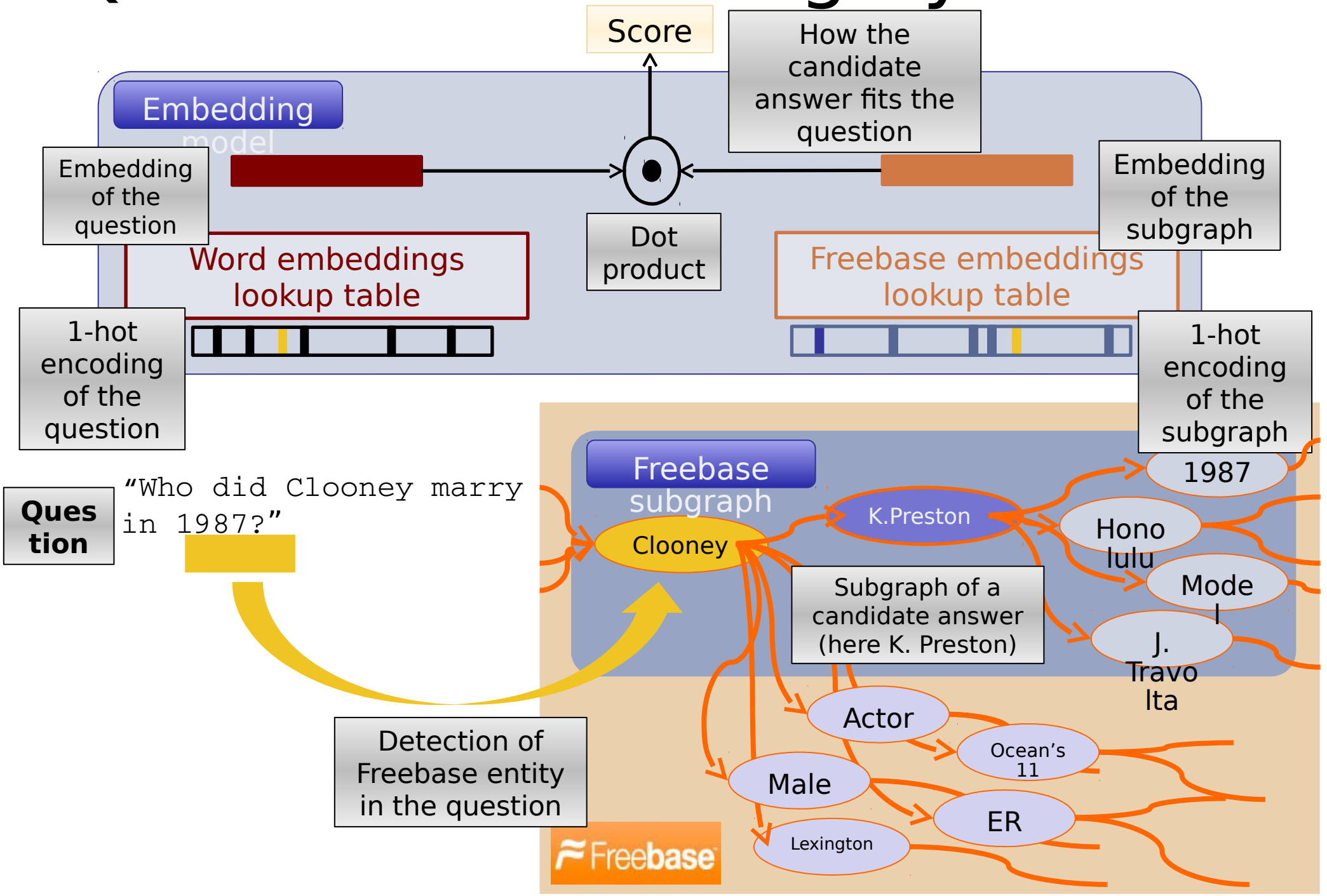


Embedding Text (with convolutional or recurrent nets)

- Embedding sentences into vector spaces
  - Using a convolutional net or a recurrent net.



Question-Answering System



Question-Answering System

what are bigos?
["stew"] ["stew"]
what are dallas cowboys colors?
["navy_blue", "royal_blue", "blue", "white", "silver"] ["blue", "navy_blue", "white", "royal_blue", "silver"]
how is egyptian money called?
["egyptian_pound"] ["egyptian_pound"]
what are fun things to do in sacramento ca?
["sacramento_zoo"] ["raging_waters_sacramento", "sutter_s_fort", "b_street_theatre", "sacramento_zoo", "california_state_capitol_museum", ...]
how are john terry's children called?
["georgie_john_terry", "summer_rose_terry"] ["georgie_john_terry", "summer_rose_terry"]
what are the major languages spoken in greece?
["greek_language", "albanian_language"] ["greek_language", "albanian_language"]
what was laura ingalls wilder famous for?
["writer", "author"] ["writer", "journalist", "teacher", "author"]

NLP: Question-Answering System

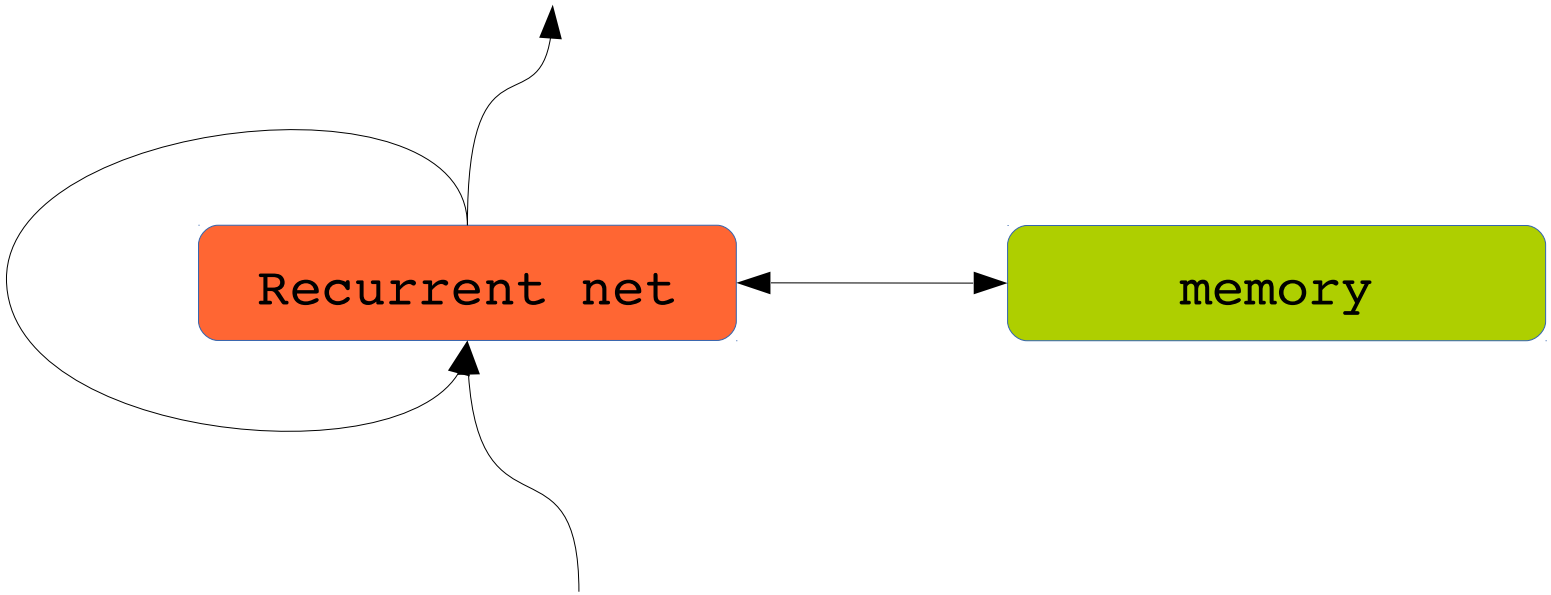
who plays sheldon cooper mother on the big bang theory?
["jim_parsons"] ["jim_parsons"]
who does peyton manning play football for?
["denver_broncos"] ["indianapolis_colts", "denver_broncos"]
who did vladimir lenin marry?
["nadezhda_krupskaya"] ["nadezhda_krupskaya"]
where was teddy roosevelt's house?
["new_york_city"] ["manhattan"]
who developed the tcp ip reference model?
["vint_cerf", "robert_e._kahn"] ["computer_scientist", "engineer”]

Representing the world with “thought vectors”

- **Every object, concept or “thought” can be represented by a vector**
  - [-0.2, 0.3, -4.2, 5.1, …..] represent the concept “cat”
  - [-0.2, 0.4, -4.0, 5.1, …..] represent the concept “dog”
  - The vectors are similar because cats and dogs have many properties in common
- **Reasoning consists in manipulating thought vectors**
  - Comparing vectors for question answering, information retrieval, content filtering
  - Combining and transforming vectors for reasoning, planning, translating languages
- **Memory stores thought vectors**
  - MemNN (Memory Neural Network) is an example
- **At FAIR we want to “embed the world” in thought vectors**

But How can Neural Nets Remember Things?

- Recurrent networks cannot remember things for very long
  - The cortex only remember things for 20 seconds
- We need a "hippocampus" (a separate memory module)
  - LSTM [Hochreiter 1997], registers
  - **Memory networks** [Weston et 2014] (FAIR), associative memory
  - NTM [DeepMind 2014], "tape".



Memory Network [Weston, Chopra, Bordes 2014]

- Add a short-term memory to a network

http://arxiv.org/abs/1410.3916

I: (input feature map) – converts the incoming input to the internal feature representation.
G: (generalization) – updates old memories given the new input.
O: (output feature map) – produces a new output (in the feature representation space), given the new input and the current memory.
R: (response) – converts the output into the response format desired. For example, a textual response or an action.

Method	F1
(Fader et al., 2013) [4]	0.54
(Bordes et al., 2014) [3]	0.73
MemNN	0.71
MemNN (with BoW features)	0.79

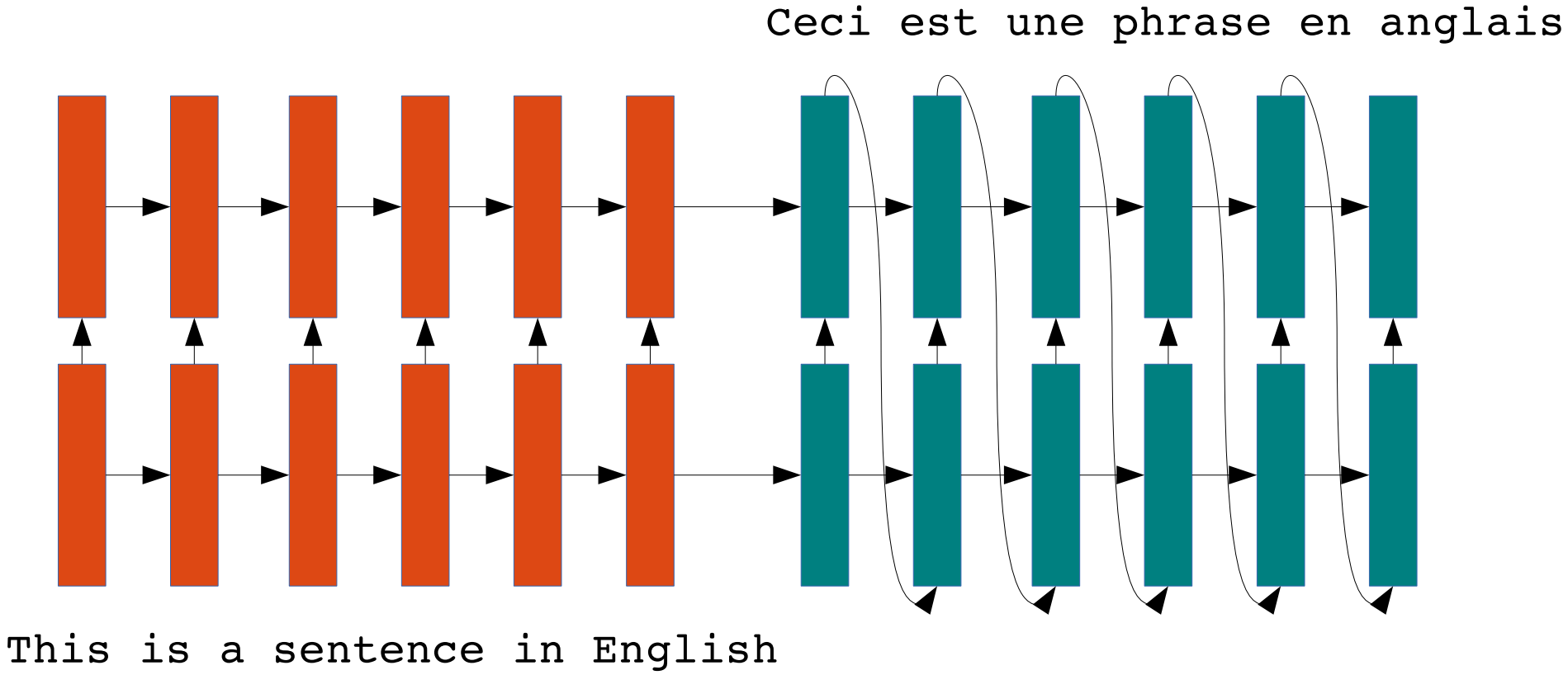
Results on Question Answering Task

Bilbo travelled to the cave.
Gollum dropped the ring there.
Bilbo took the ring.
Bilbo went back to the Shire.
Bilbo left the ring there.
Frodo got the ring.
Frodo journeyed to Mount-Doom.
Frodo dropped the ring there.
Sauron died.
Frodo went back to the Shire.
Bilbo travelled to the Grey-havens.
The End.
Where is the ring? A: Mount-Doom
Where is Bilbo now? A: Grey-havens
Where is Frodo now? A: Shire

Fig. 2. An example story with questions correctly answered by a MemNN. The MemNN was trained on the simulation described in Section 4.2 and had never seen many of these words before, e.g. Bilbo, Frodo and Gollum.

Language Translation with LSTM networks

- [Sutskever et al. NIPS 2014]
  - Multiple layers of very large LSTM recurrent modules
  - English sentence is read in and encoded
  - French sentence is produced after the end of the English sentence
  - Accuracy is very close to state of the art.



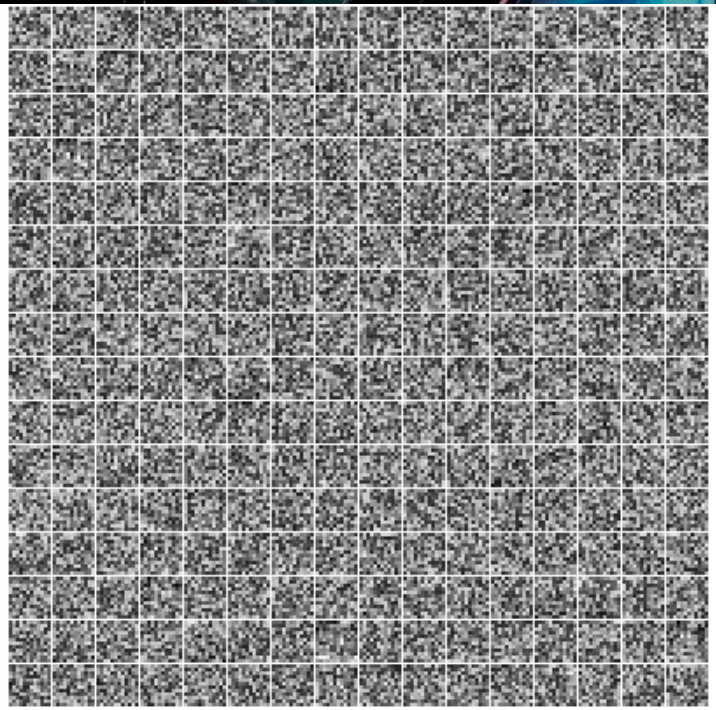
Language Translation with LSTM networks

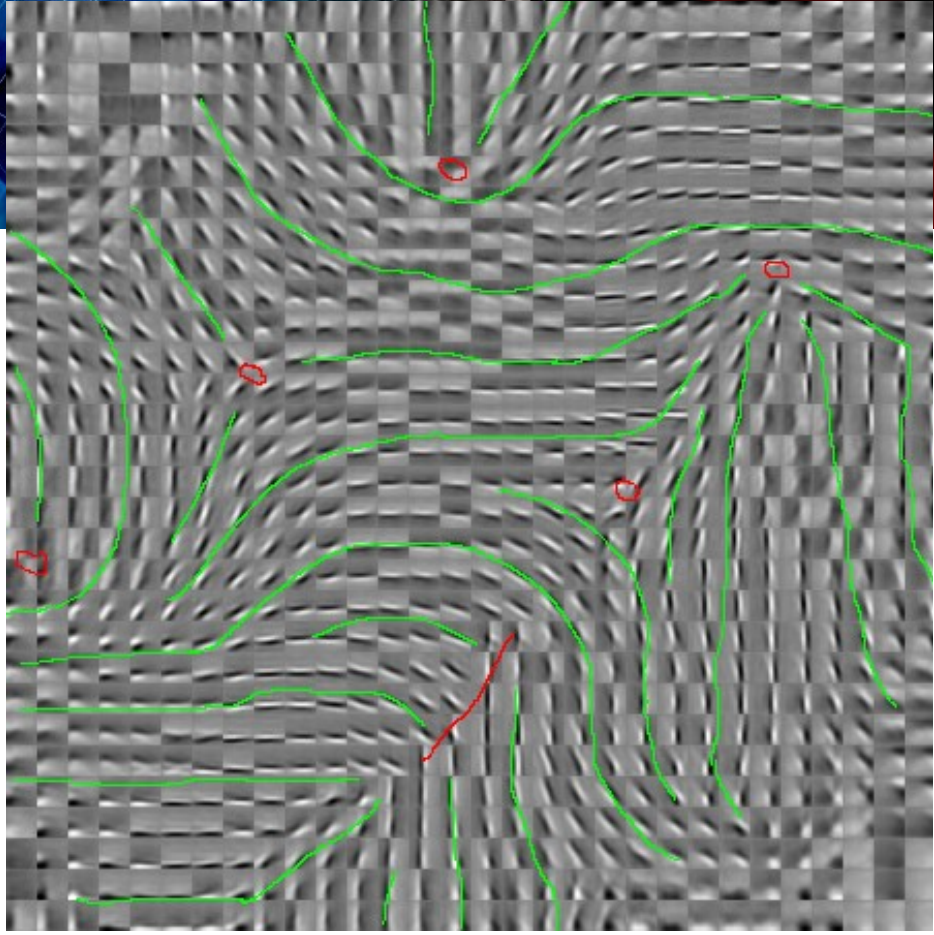
- [Sutskever et al. NIPS 2014]
  - Multiple layers of very large LSTM recurrent modules
  - English sentence is read in and encoded
  - French sentence is produced after the end of the English sentence
  - Accuracy is very close to state of the art.

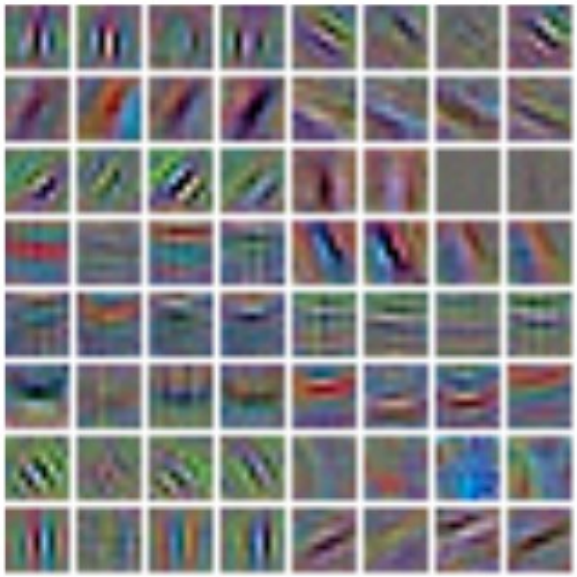
Our model	Ulrich UNK , membre du conseil d' administration du constructeur automobile Audi , affirme qu' il s' agit d' une pratique courante depuis des années pour que les téléphones portables puissent être collectés avant les réunions du conseil d' administration afin qu' ils ne soient pas utilisés comme appareils d' écoute à distance .
Truth	Ulrich Hackenberg , membre du conseil d' administration du constructeur automobile Audi , déclare que la collecte des téléphones portables avant les réunions du conseil , afin qu' ils ne puissent pas être utilisés comme appareils d' écoute à distance , est une pratique courante depuis des années .


Unsupervised Learning of
Invariant Features

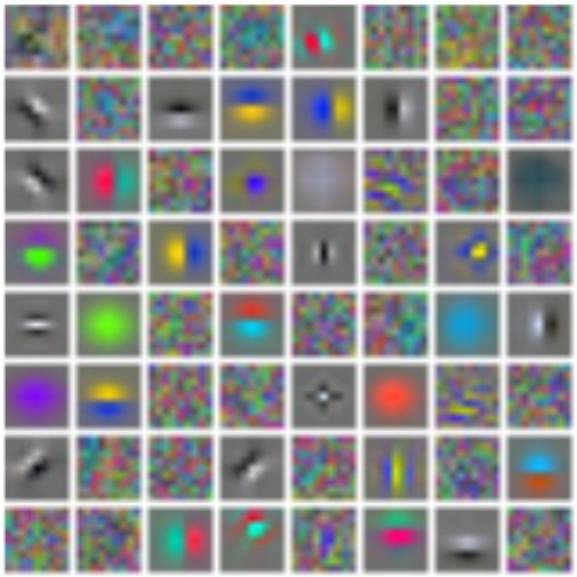
Unsupervised Learning



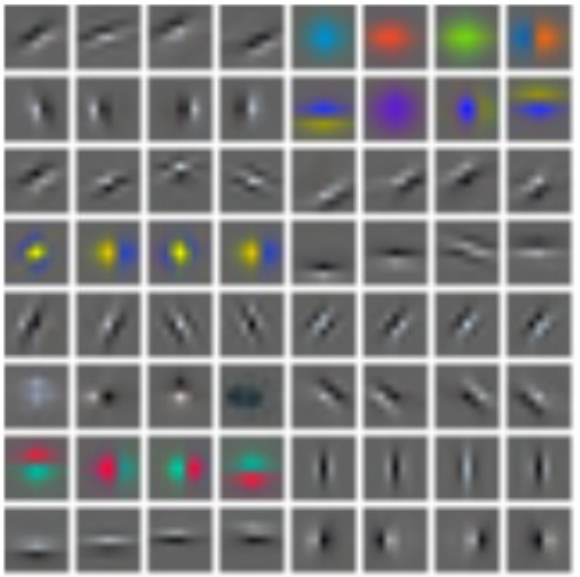




Supervised filters



sparse convolutional AE



slow & sparse convolutional AE

Unsupervised Learning by Prediction and Linearization

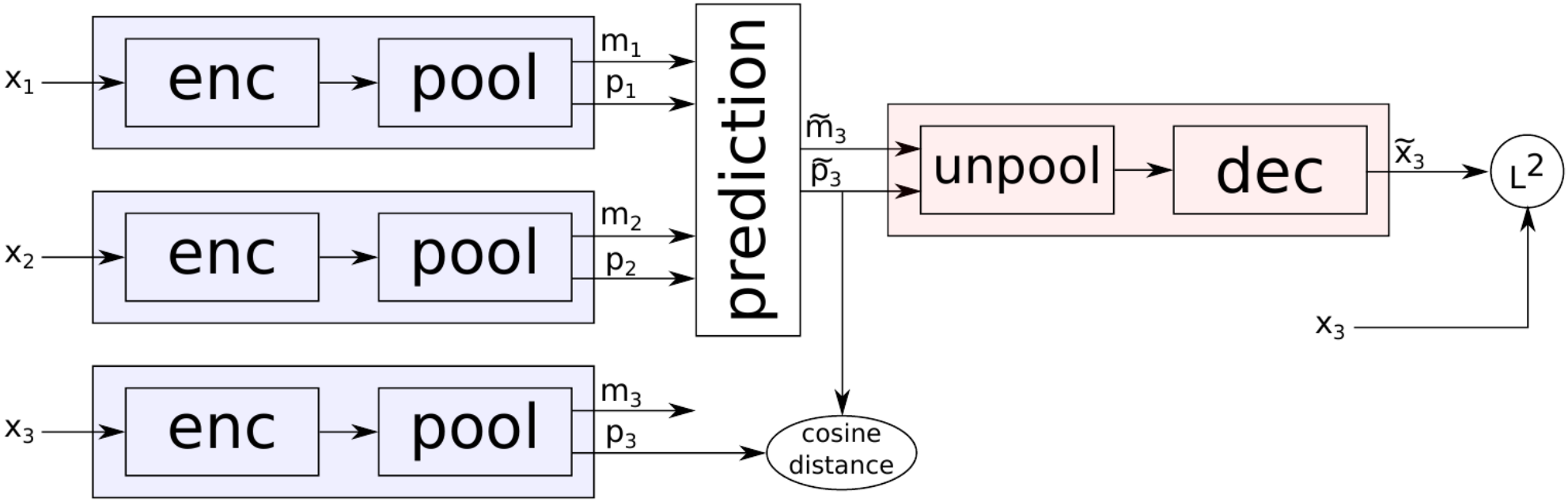
$$L = \frac{1}{2}\|G_W(\mathbf{a}\begin{bmatrix} z^t & z^{t-1}\end{bmatrix}^T) - x^{t+1}\|_2^2 - \lambda \frac{(z^t - z^{t-1})^T(z^{t+1} - z^t)}{\|z^t - z^{t-1}\|\|z^{t+1} - z^t\|}$$

- **Magnitude**
  - (soft max)

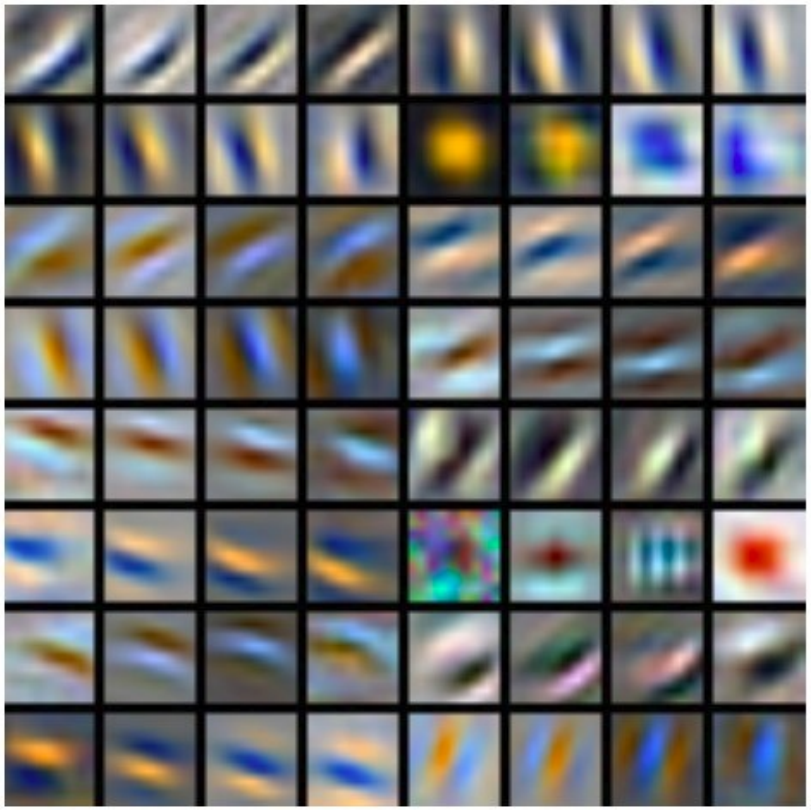
$$m_k = \sum_{N_k} z(f,x,y) \frac{e^{\beta z(f,x,y)}}{\sum_{N_k} e^{\beta z(f\prime,x\prime,y\prime)}} \approx \max_{N_k} z(f,x,y)$$

- **Phase**
  - (soft argmax)

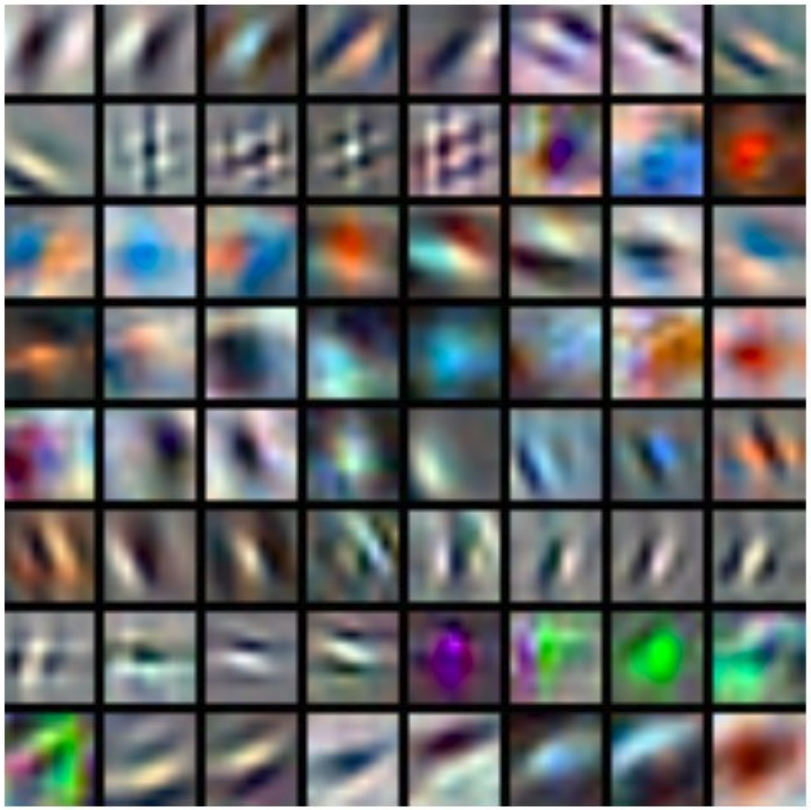
$$\mathbf{p}_k = \sum_{N_k} \begin{bmatrix} f \\ x \\ y \end{bmatrix} \frac{e^{\beta z(f,x,y)}}{\sum_{N_k} e^{\beta z(f\prime,x\prime,y\prime)}} \approx \arg\max_{N_k} z(f,x,y)$$



Filters: pooling over groups of 4 filters.



(a) Shallow Architecture 1



(b) Shallow Architecture 2

Take Home Messages

- **Deep Learning is enabling a new wave of applications**
  - **Today:** Image recognition, video understanding: vision now works
  - **Today:** Better speech recognition: speech recognition now works
  - **Soon:** Better language understanding, dialog, and translation
- **Deep Learning and Convolutional Nets are widely deployed**
  - **Today:** image understanding at Facebook, Google, Twitter, Microsoft…..
  - **Soon:** better auto-pilots for cars, medical image analysis, robot perception
- **We need hardware (and software) for embedded applications**
  - For smart cameras, mobile devices, cars, robots, toys….
- **But we are still far from building truly intelligent machines**
  - We need to integrate reasoning with deep learning
  - We need a good architecture for "episodic" (short-term) memory.
  - We need to find good principles for unsupervised learning