

Unsupervised Learning: The Next Frontier In AI

Yann Le Cun

Facebook AI Research

Center for Data Science, NYU

Courant Institute of Mathematical Sciences, NYU

<http://yann.lecun.com>





1st Connectionist Summer School, CMU July 1986

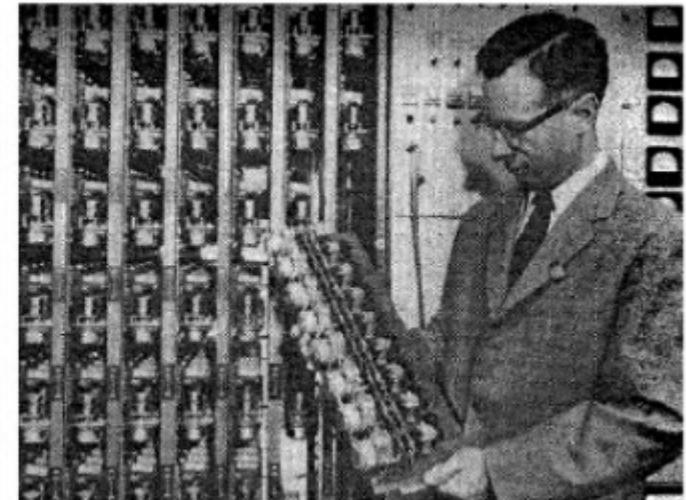
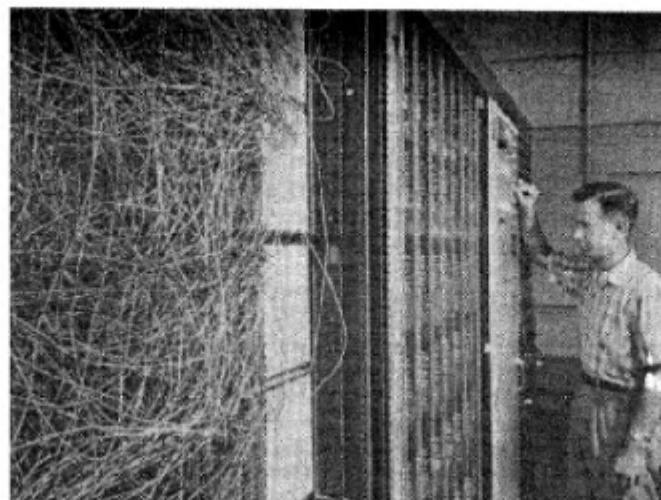
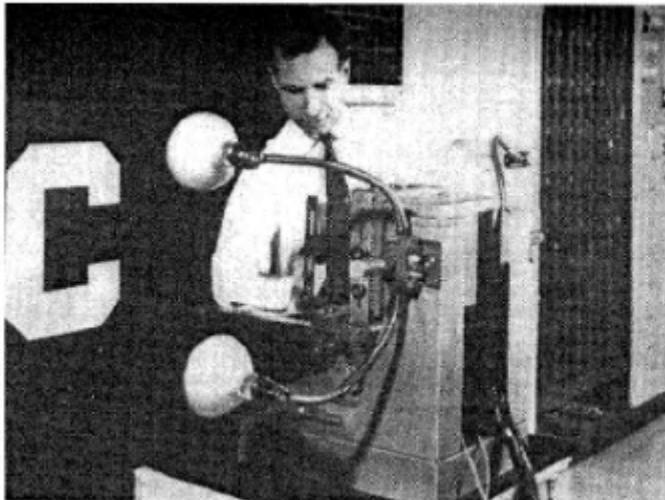
Y LeCun



Neuroscience inspired early works on Machine Learning & AI

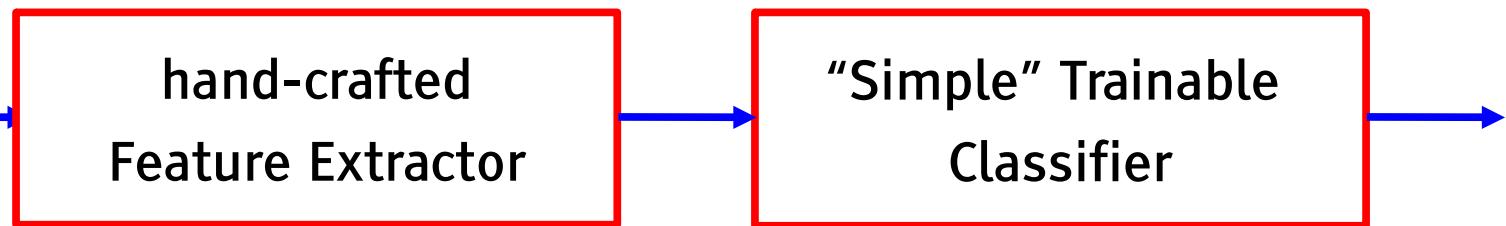
Y LeCun

The Perceptron (Frank Rosenblatt at Cornell University, 1957)



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

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



Supervised Learning

- We can train a machine on lots of examples of tables, chairs, dog, cars, and people
- But will it recognize table, chairs, dogs, cars, and people it has never seen before?



PLANE



CAR



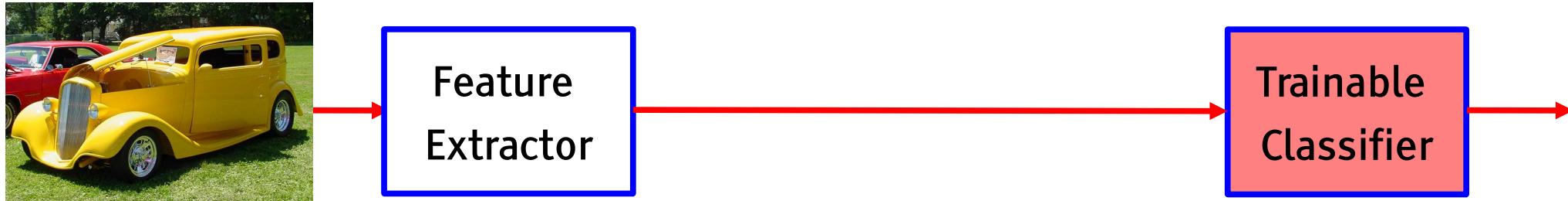
CAR



Deep Learning = The Entire Machine is Trainable

Y LeCun

■ Traditional Pattern Recognition: Fixed/Handcrafted Feature Extractor



■ Mainstream Modern Pattern Recognition: Unsupervised mid-level features



■ Deep Learning: Representations are hierarchical and trained

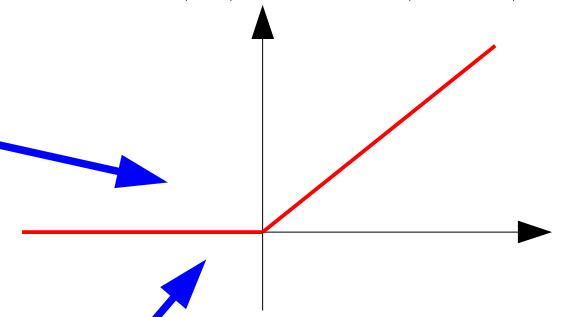


Multi-Layer Neural Nets

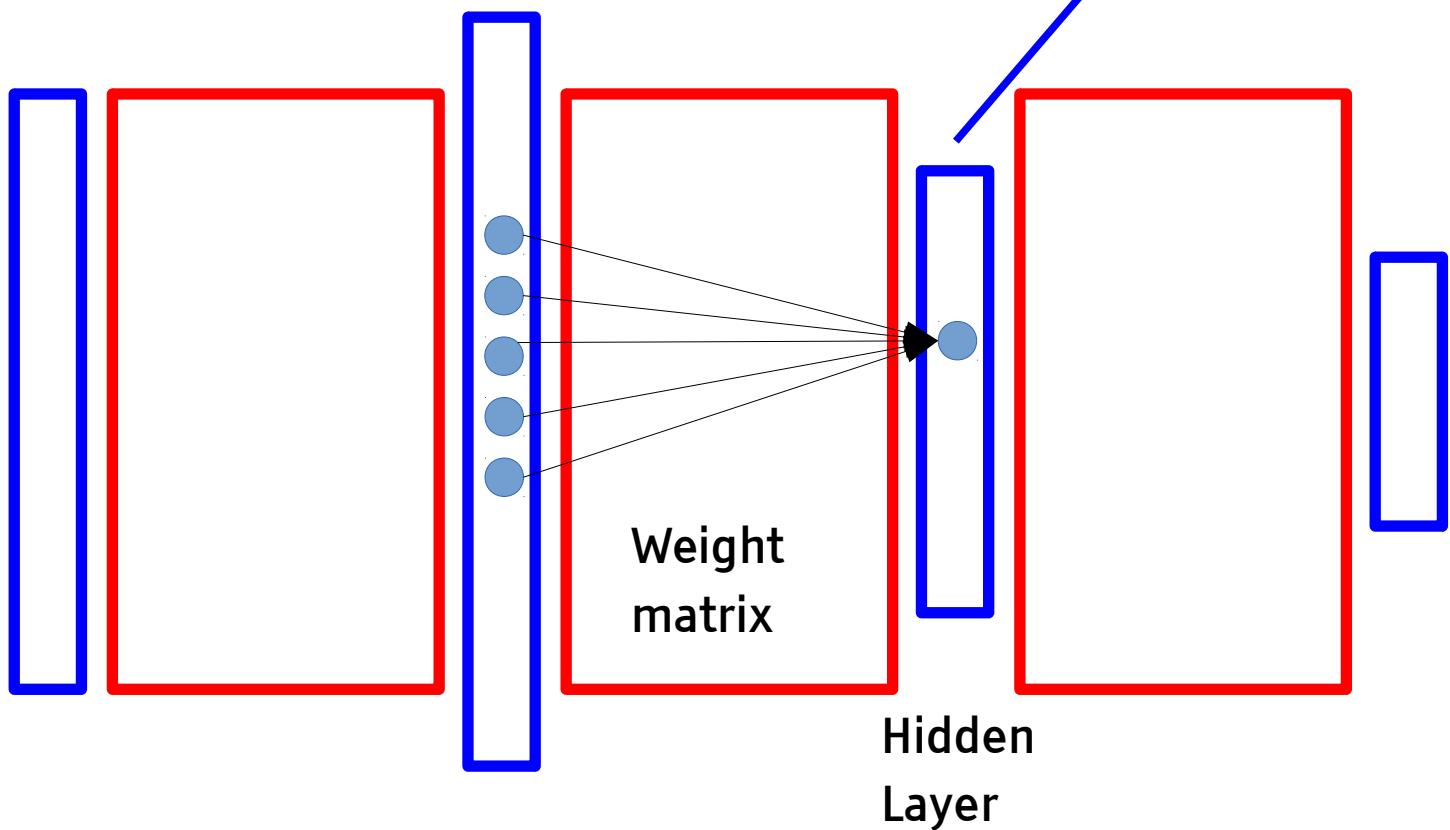
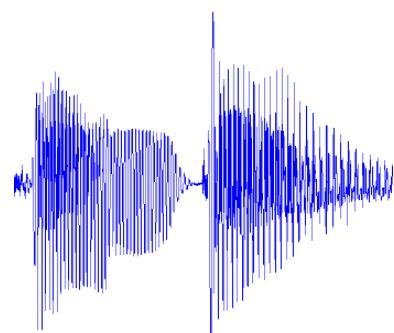
Y LeCun

- Multiple Layers of **simple units**
- Each units computes a **weighted sum** of its inputs
- Weighted sum is passed through a **non-linear function**
- The learning algorithm changes the **weights**

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

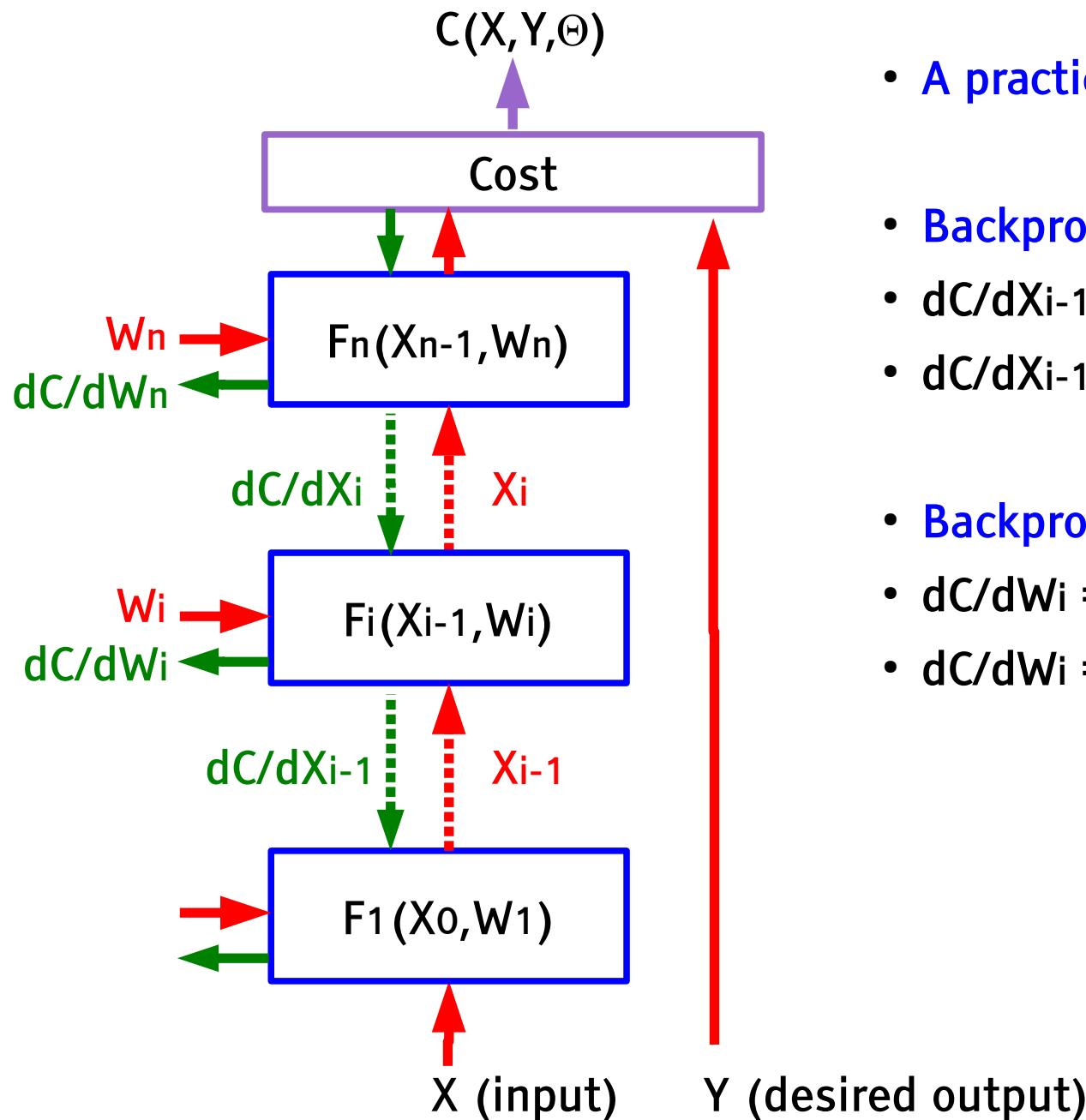


Ceci est une voiture



Computing Gradients by Back-Propagation

Y LeCun



- A practical Application of Chain Rule

- Backprop for the state gradients:

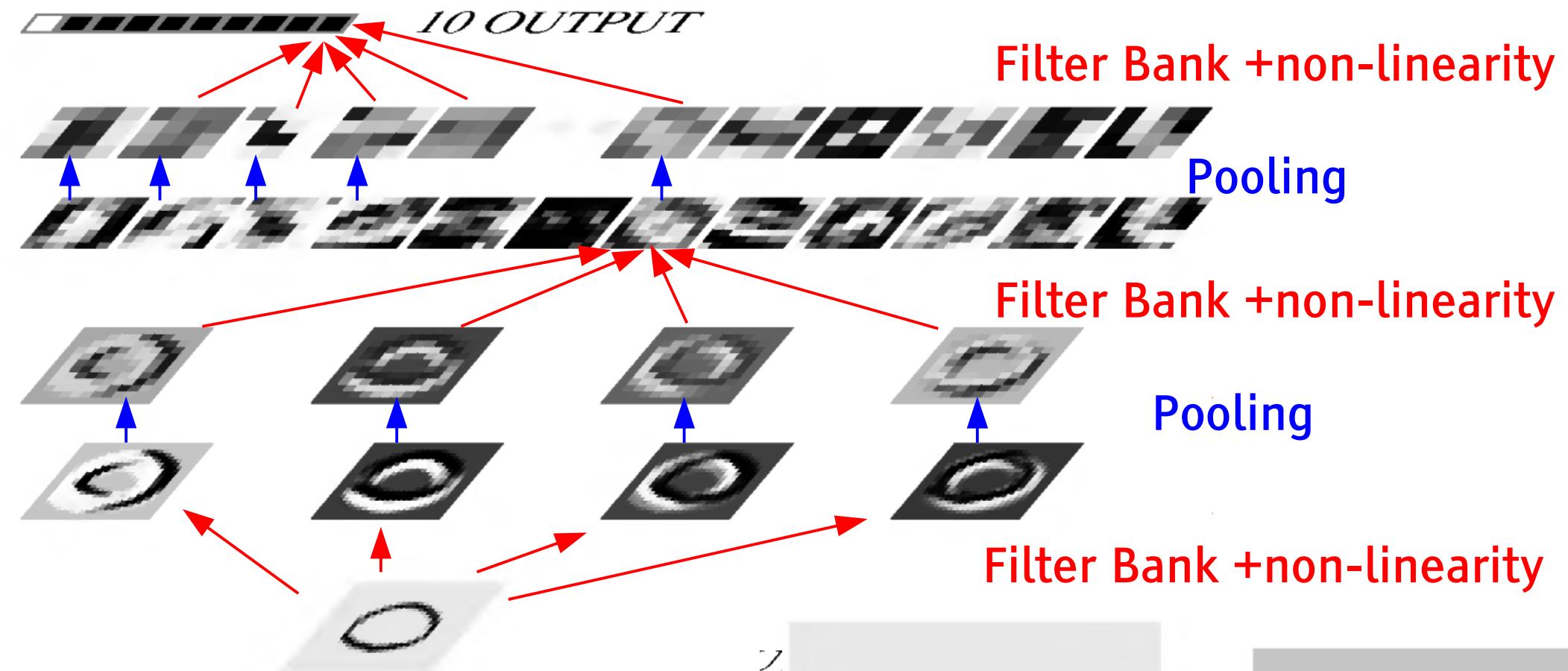
- $dC/dX_{i-1} = dC/dX_i \cdot dX_i/dX_{i-1}$
- $dC/dX_i = dC/dX_i \cdot dF_i(X_{i-1}, W_i)/dX_i$

- Backprop for the weight gradients:

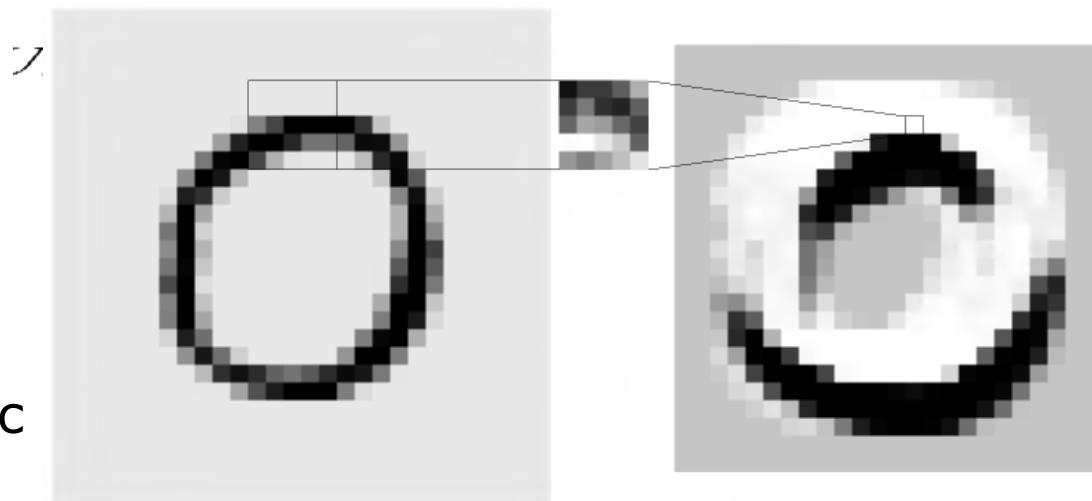
- $dC/dW_i = dC/dX_i \cdot dX_i/dW_i$
- $dC/dW_i = dC/dX_i \cdot dF_i(X_{i-1}, W_i)/dW_i$

Convolutional Network Architecture [LeCun et al. NIPS 1989]

Y LeCun



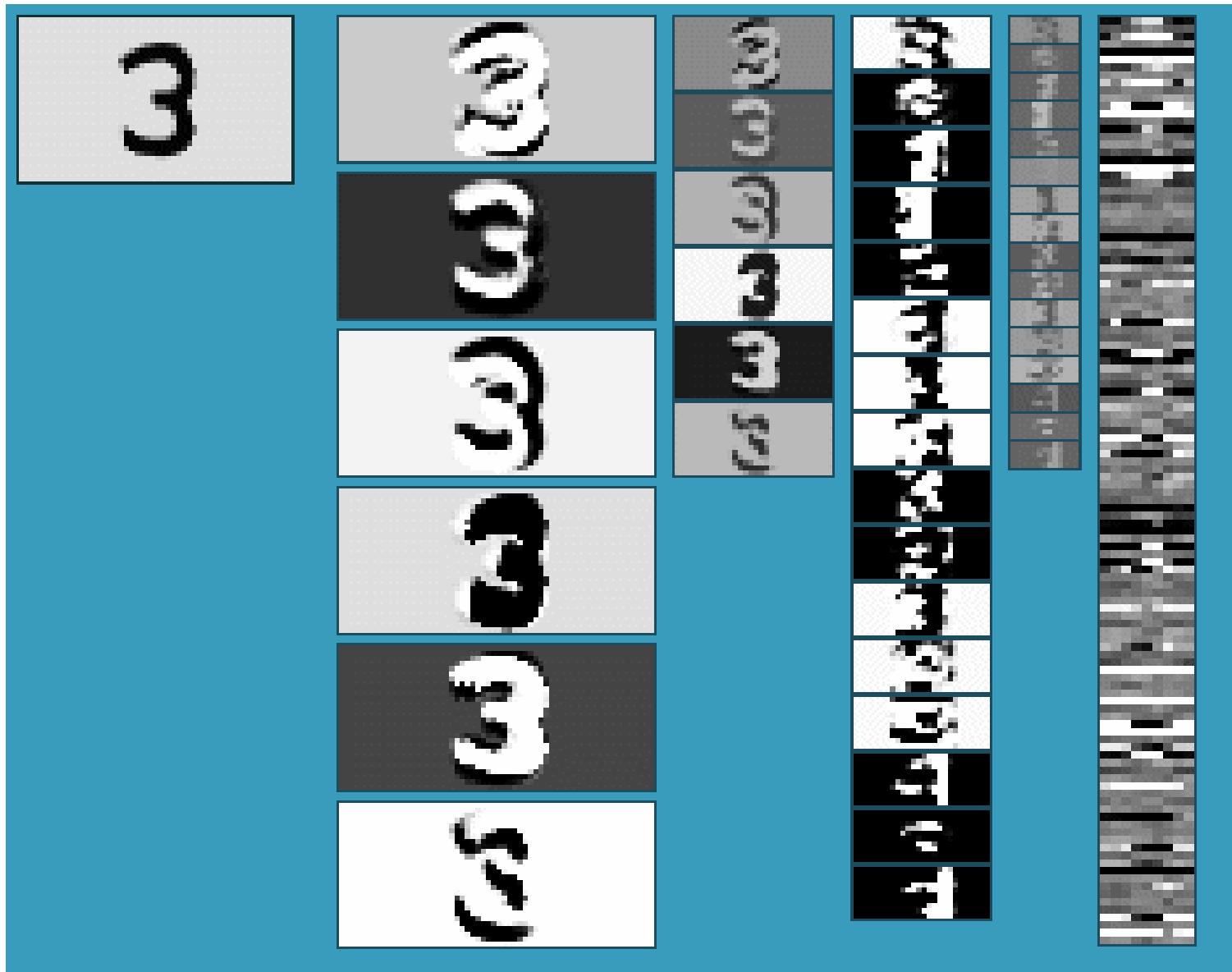
- Inspired by [Hubel & Wiesel 1962] & [Fukushima 1982] (Neocognitron):
 - ▶ simple cells detect local features
 - ▶ complex cells “pool” the outputs of simple cells within a retinotopic neighborhood.



Convolutional Network (vintage 1990)

Y LeCun

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

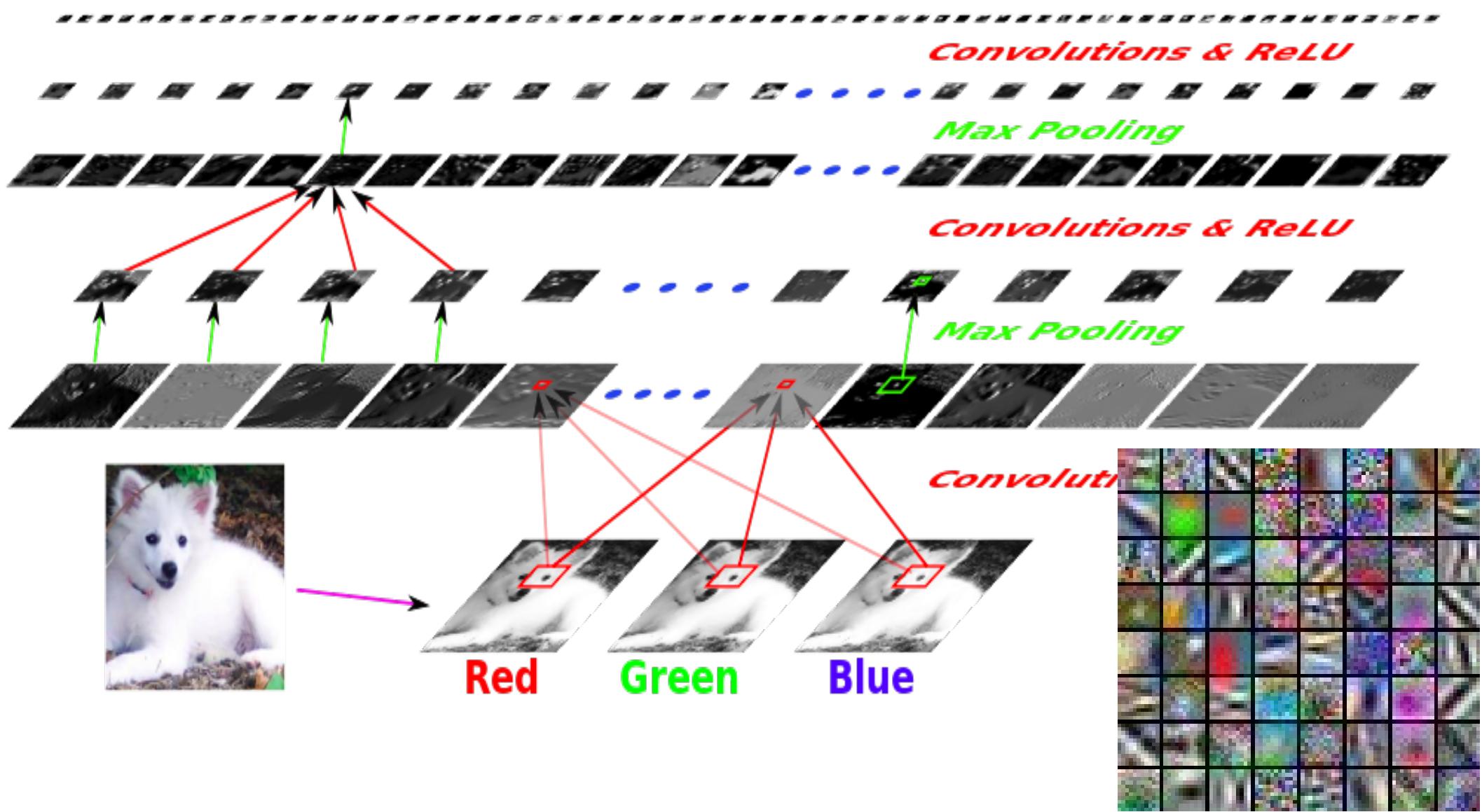




Deep Convolutional Nets for Object Recognition

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

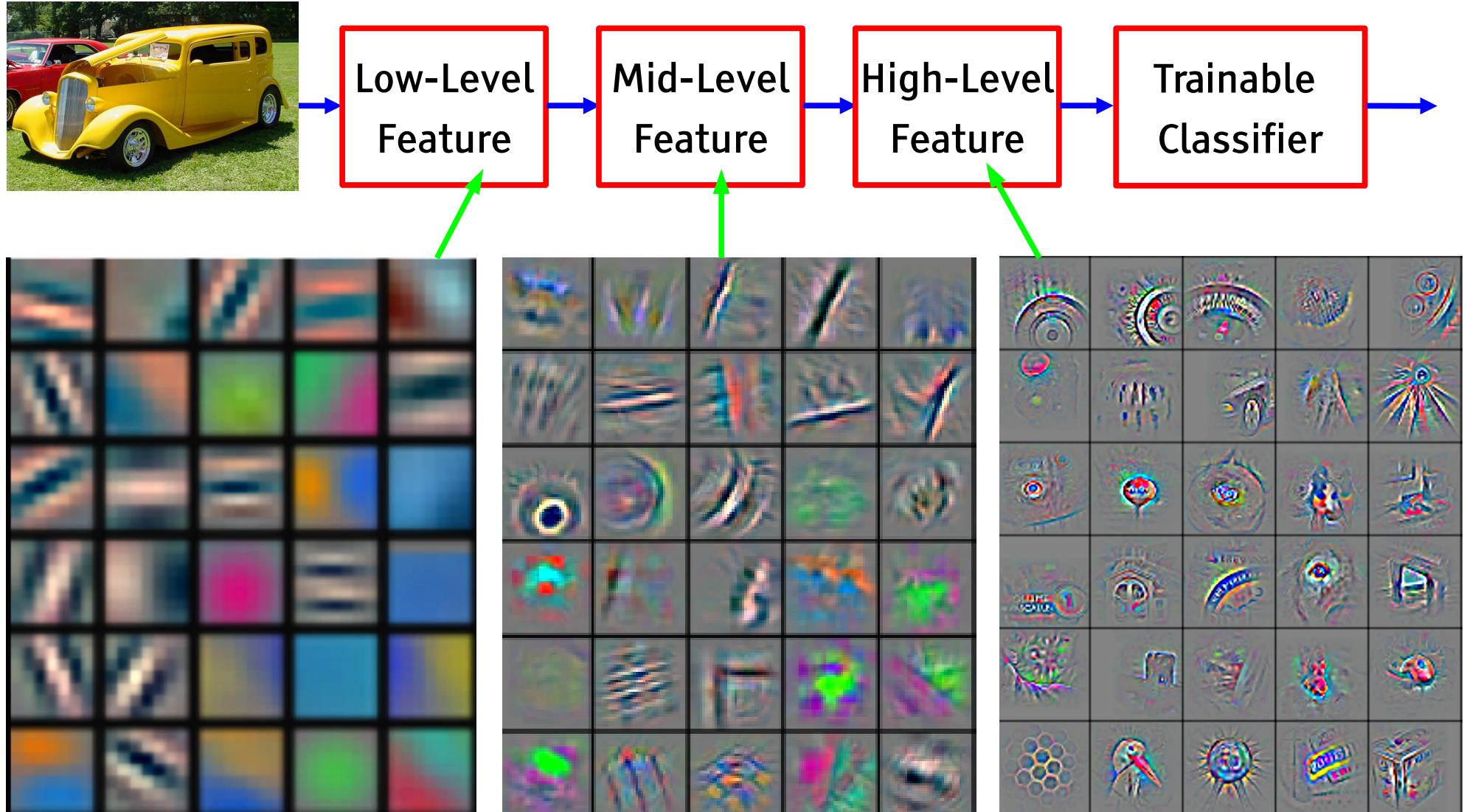
Samoyed (16); Papillon (5.7); Pomeranian (2.7); Arctic Fox (1.0); Eskimo Dog (0.6); White Wolf (0.4); Siberian Husky (0.4)



Deep Learning = Learning Hierarchical Representations

Y LeCun

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

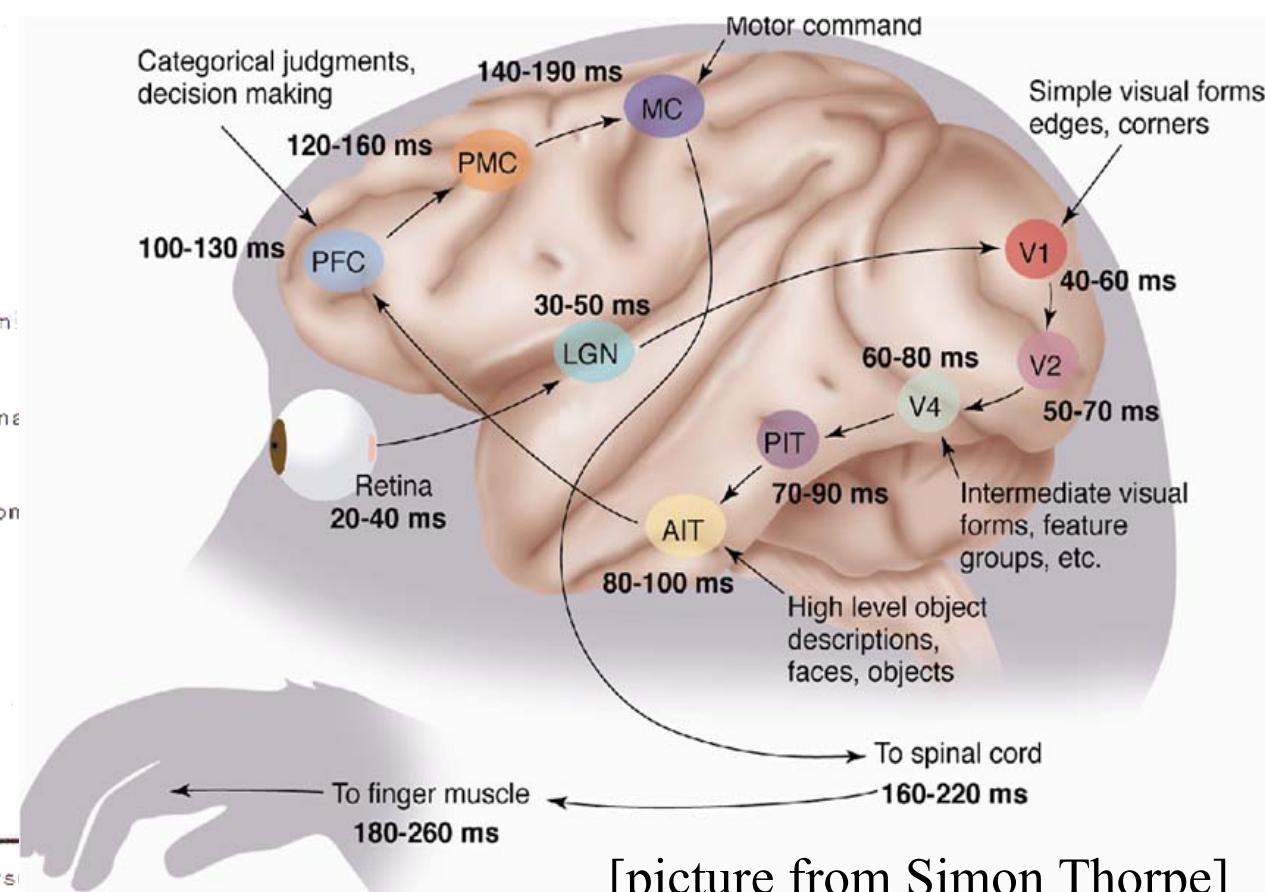
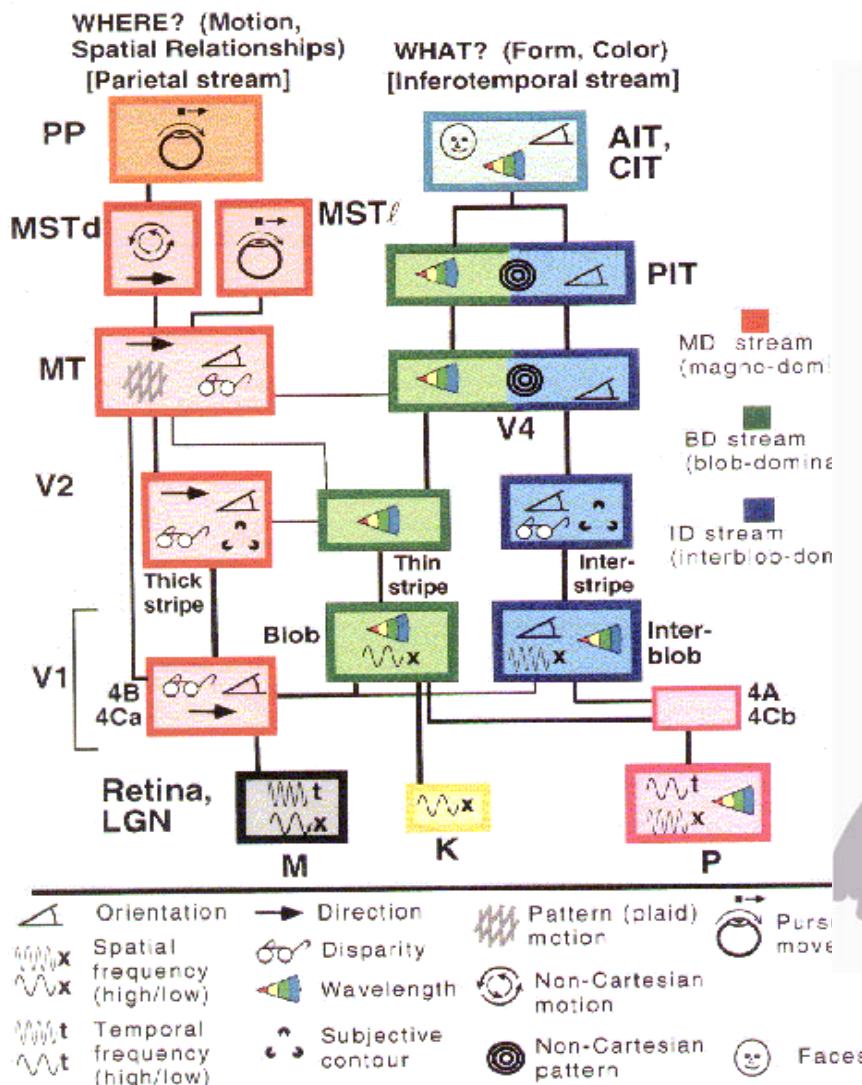


Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

Hierarchical Structure in the Visual Cortex

Y LeCun

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



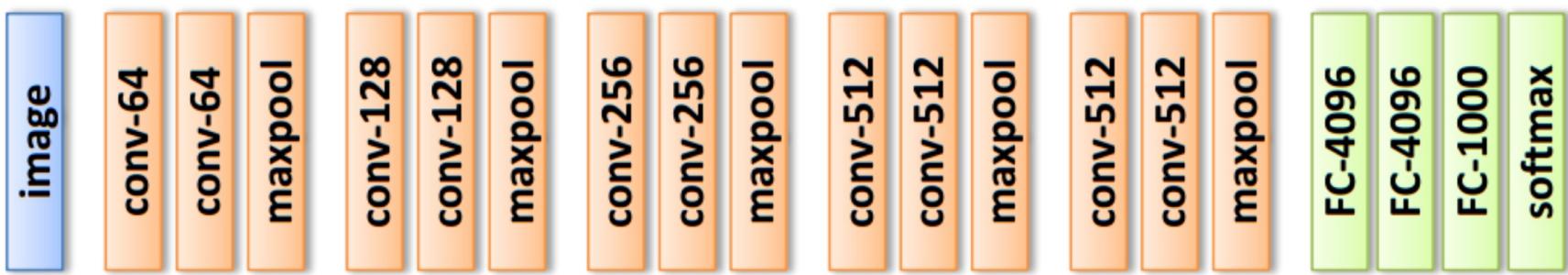
[Gallant & Van Essen]

Very Deep ConvNet Architectures

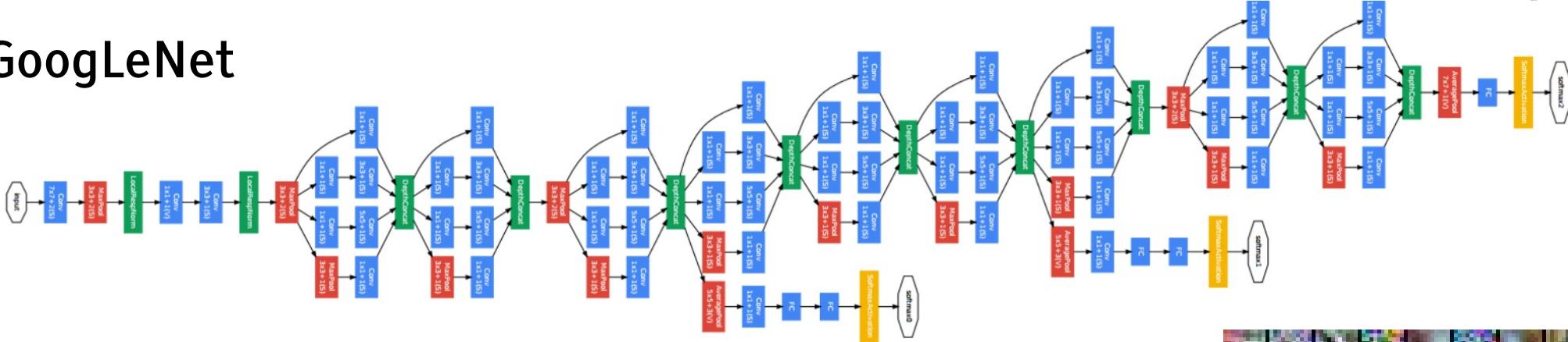
Y LeCun

Small kernels, not much subsampling (fractional subsampling).

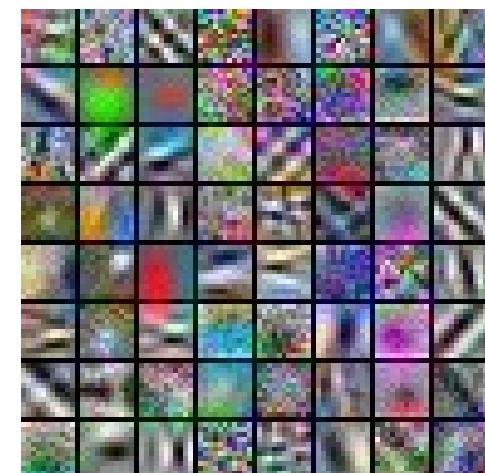
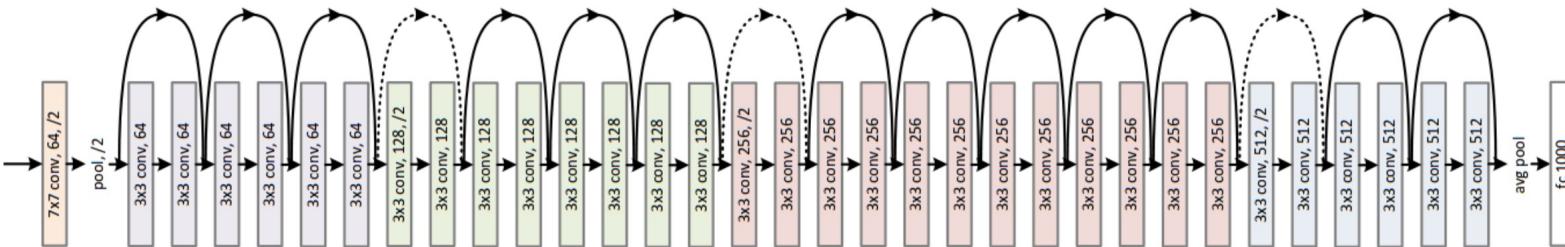
VGG



GoogLeNet



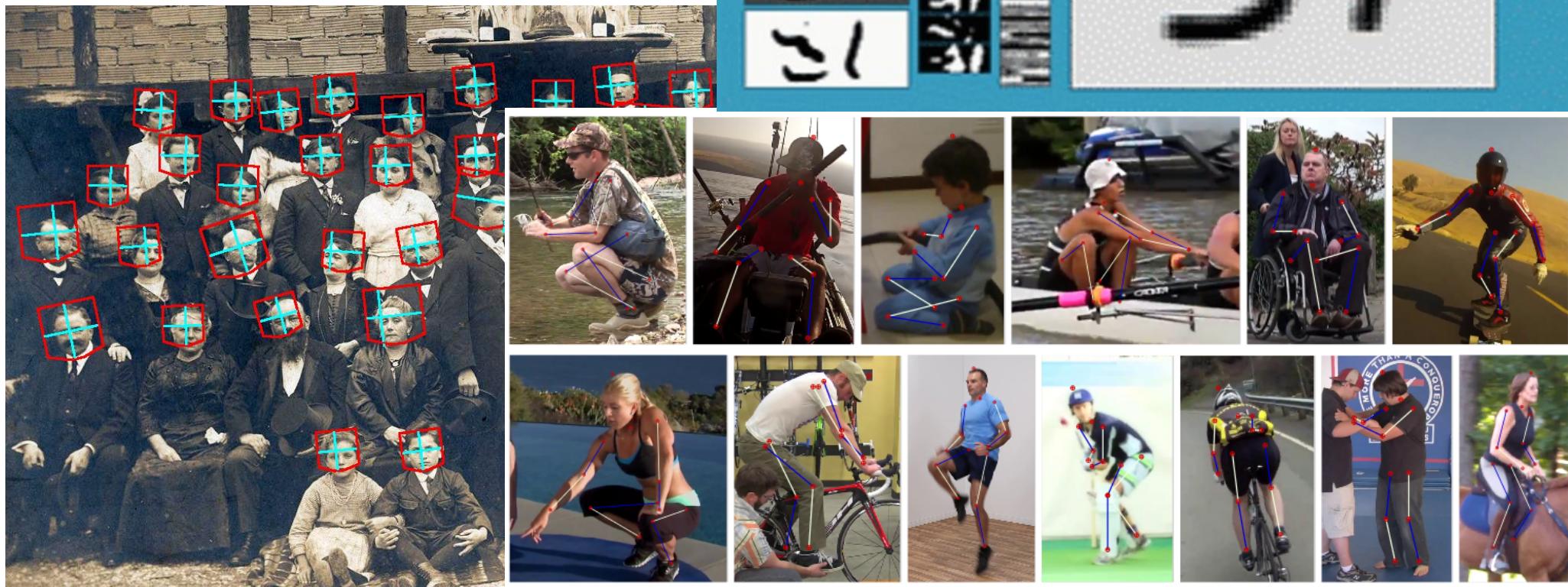
ResNet

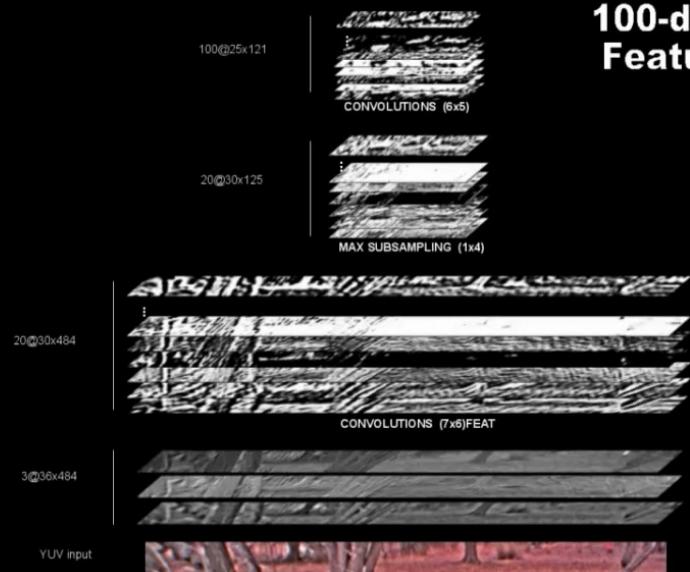


f Supervised Convolutional Nets

We can do a lot with Supervised ConvNets

- ▶ Detect and localize objects
- ▶ Recognize multiple objects
- ▶ Estimate the pose of articulated objects (human bodies)
- ▶





100-dimensional Feature Vector

ConvNet for Driving

Y LeCun

(DARPA LAGR program 2005-2008)

[Hadsell et al., J. of Field Robotics 2009]

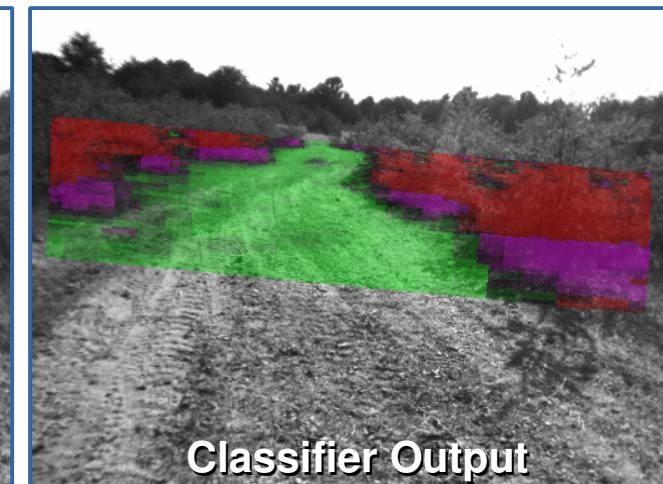
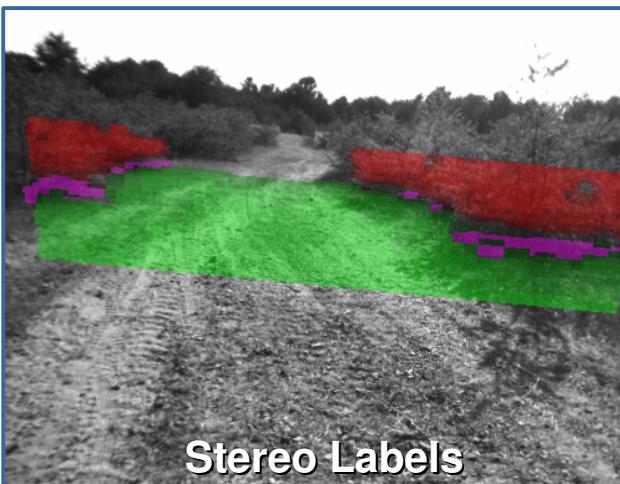
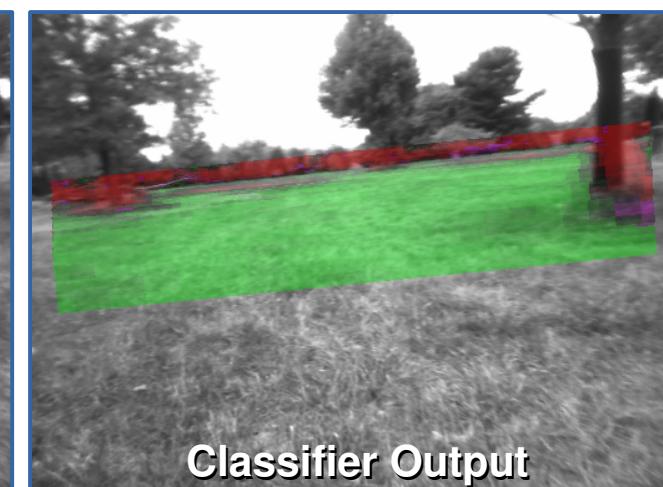
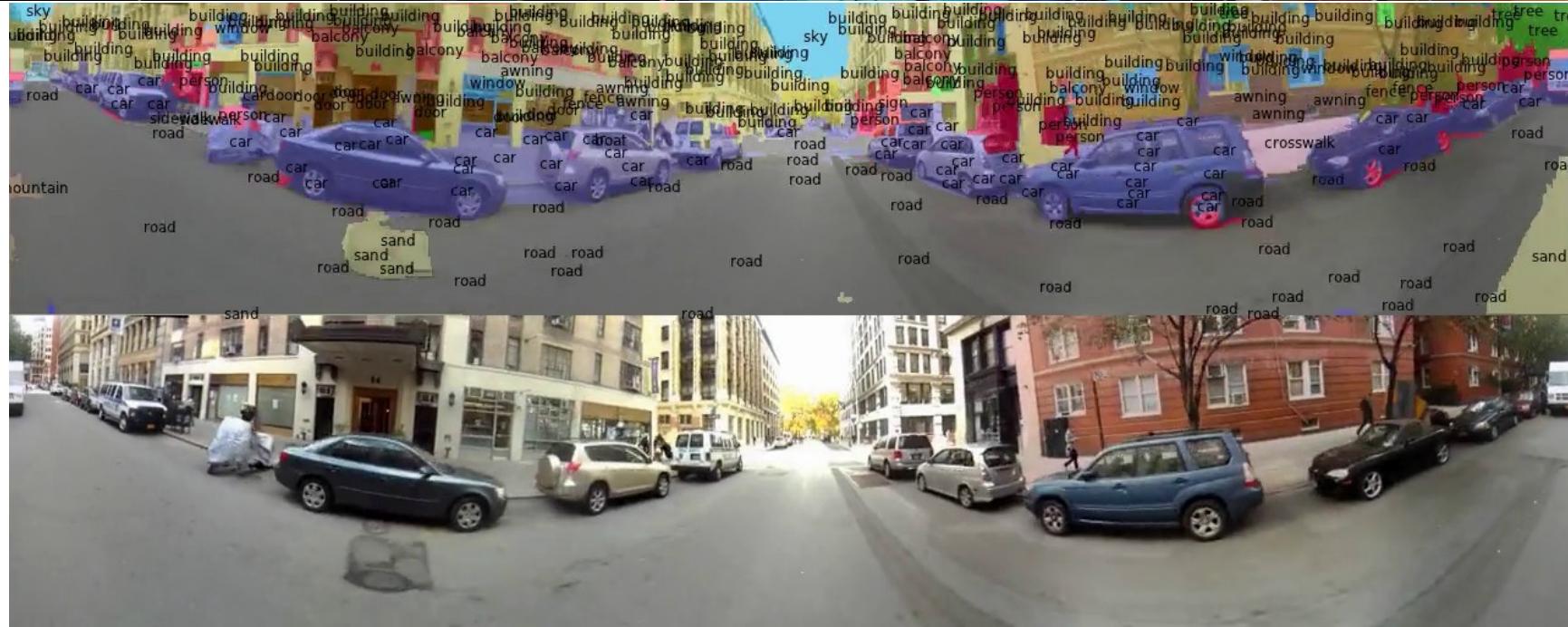


Image captioning, Semantic Segmentation with ConvNets

Y LeCun



[Farabet et al.
ICML 2011]

[Farabet et al.
PAMI 2013]



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.

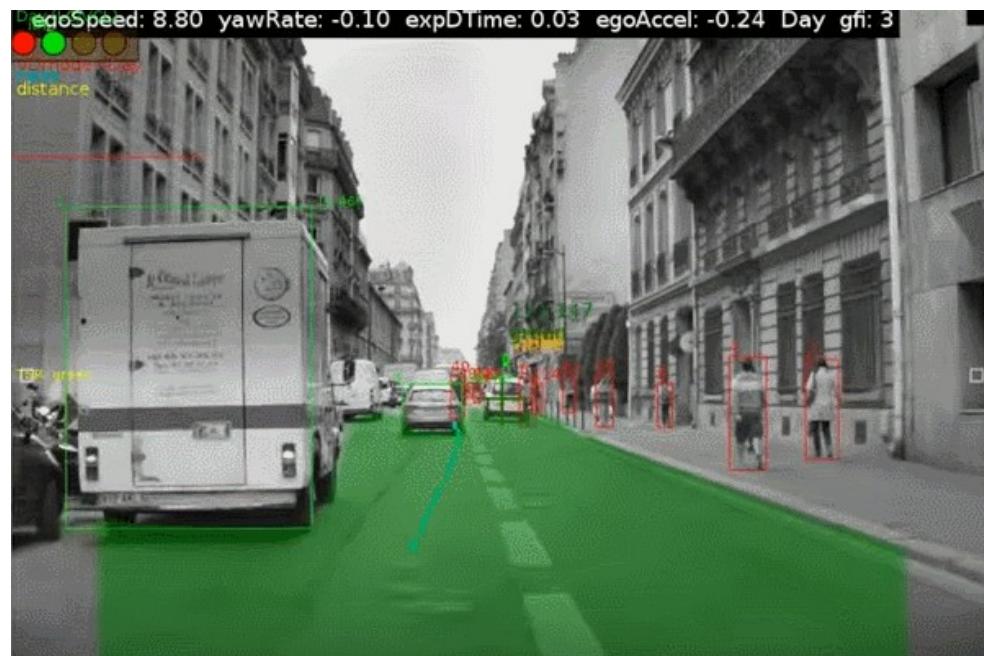
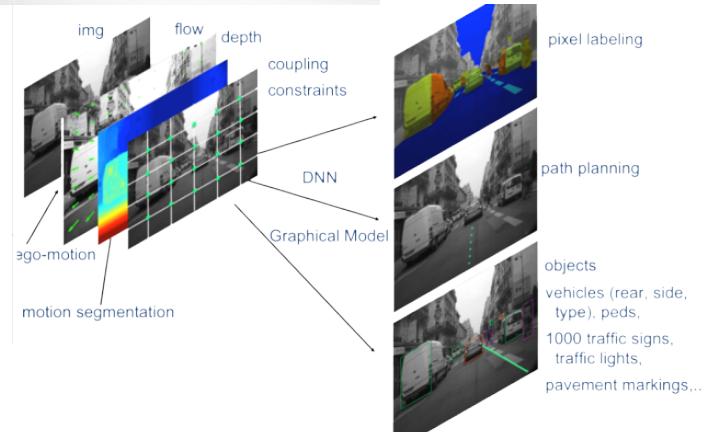


[Lebret, Pinheiro, Collobert 2015] [Kulkarni 11] [Mitchell 12] [Vinyals 14] [Mao 14] [Karpathy 14] [Donahue 14] ...

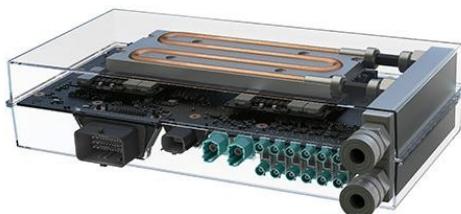
Driving Cars with Convolutional Nets

Y LeCun

MobilEye



NVIDIA

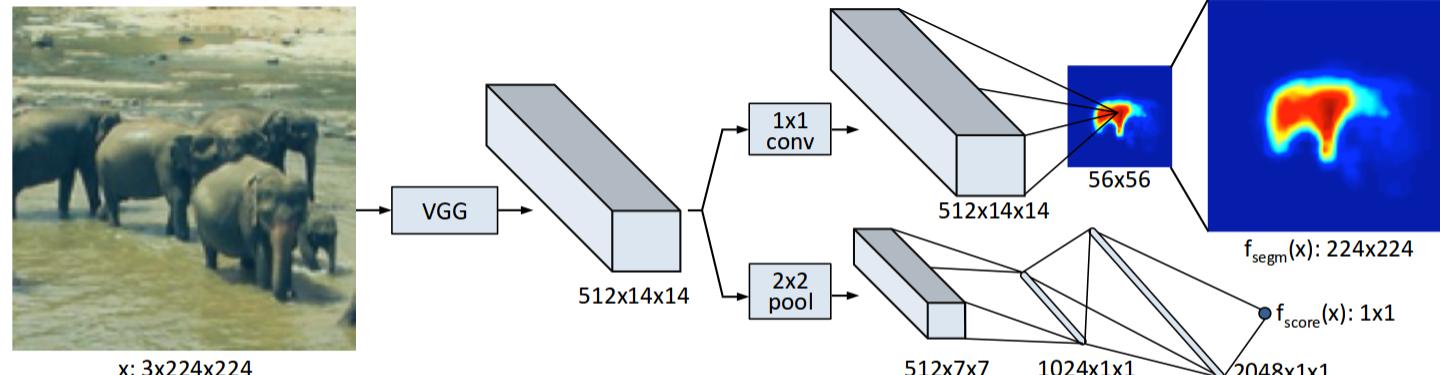


DeepMask: ConvNet Locates and Recognizes Objects

Y LeCun

[Pinheiro, Collobert,
Dollar ICCV 2015]

- ConvNet produces object masks and categories





DeepMask++ Proposals

Y LeCun

■ <https://arxiv.org/abs/1604.02135>

■ Zagoruyko, Lerer, Lin, Pinheiro, Gross, Chintala, Dollár





Image Recognition

Y LeCun





Image Recognition

Y LeCun

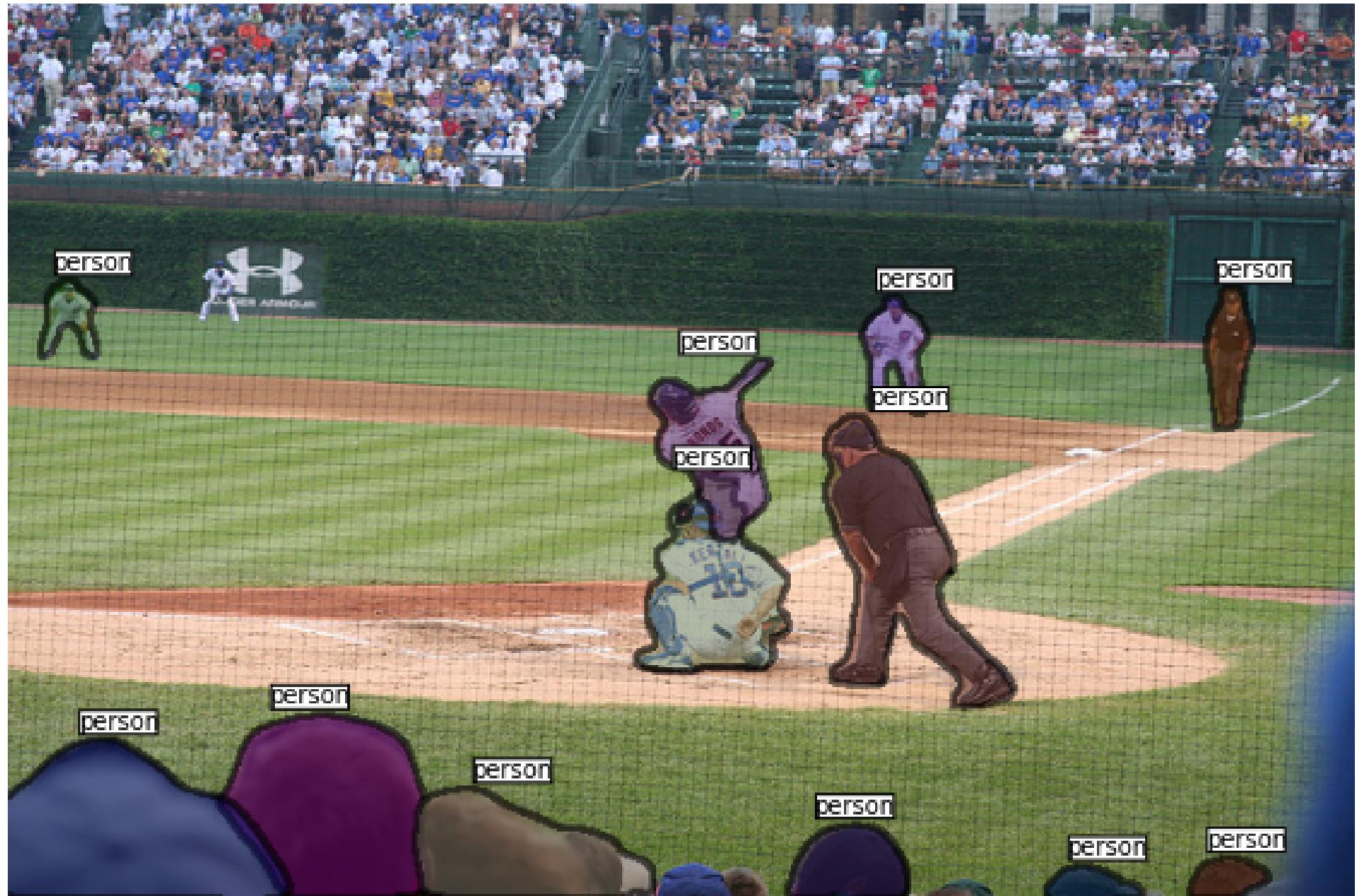




Image Recognition

Y LeCun





Image Recognition

Y LeCun



Image Recognition

Y LeCun

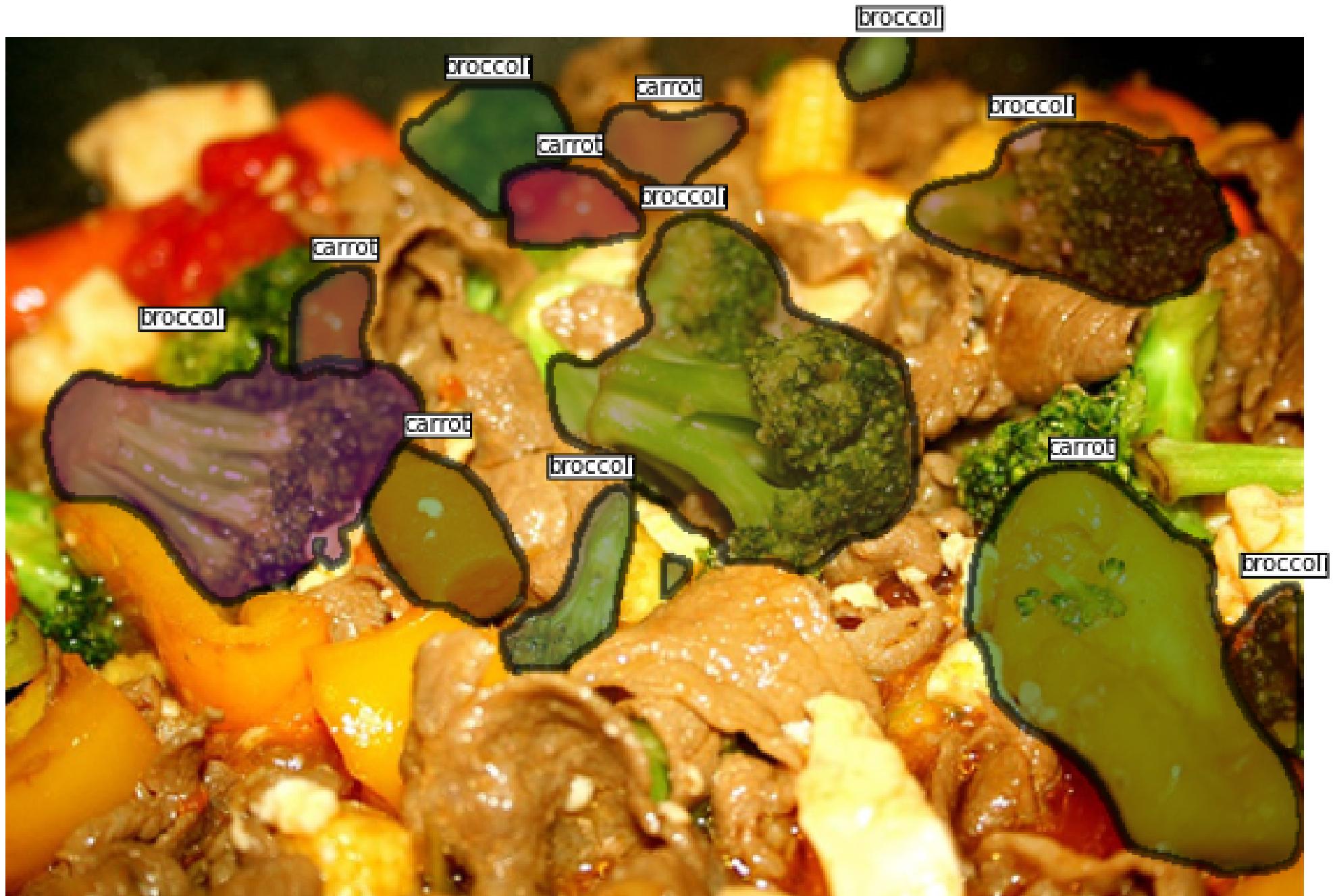




Image Recognition

Y LeCun



Results

Y LeCun

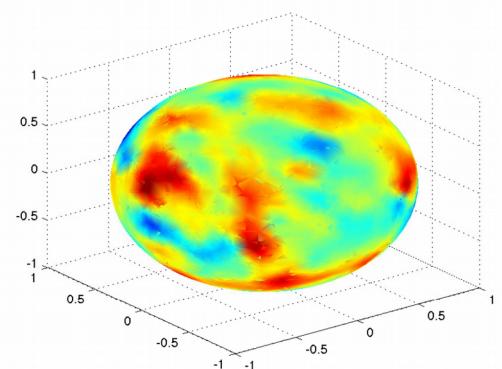
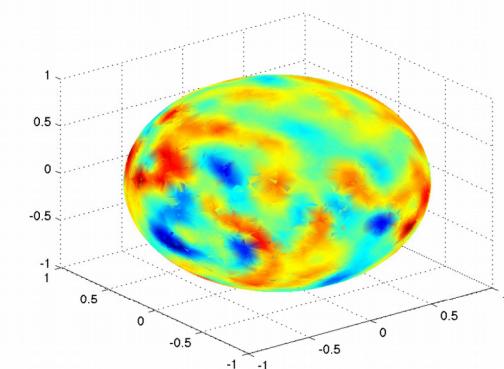
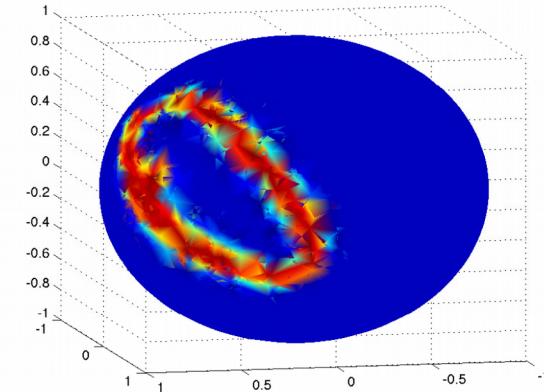
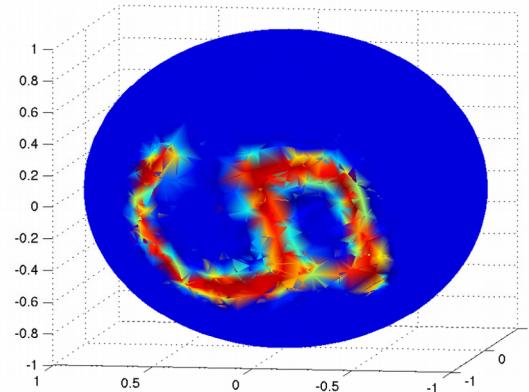


Spectral Networks Convolutional Nets on Graphs

Spectral Networks: Convolutional Nets on Irregular Graphs

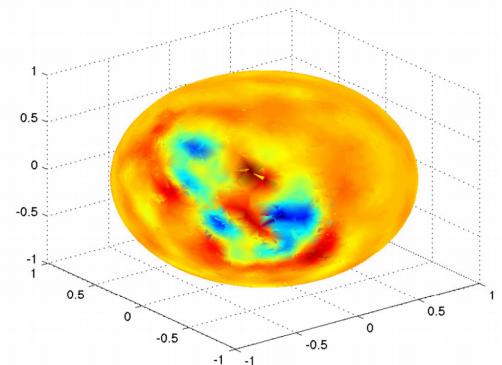
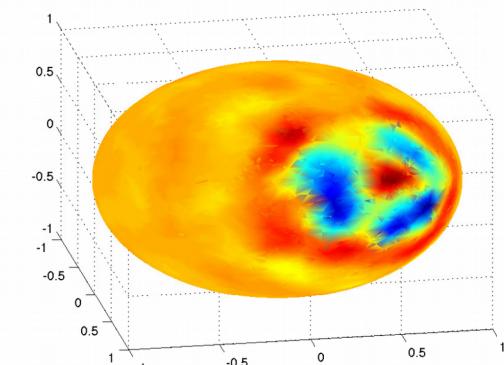
Y LeCun

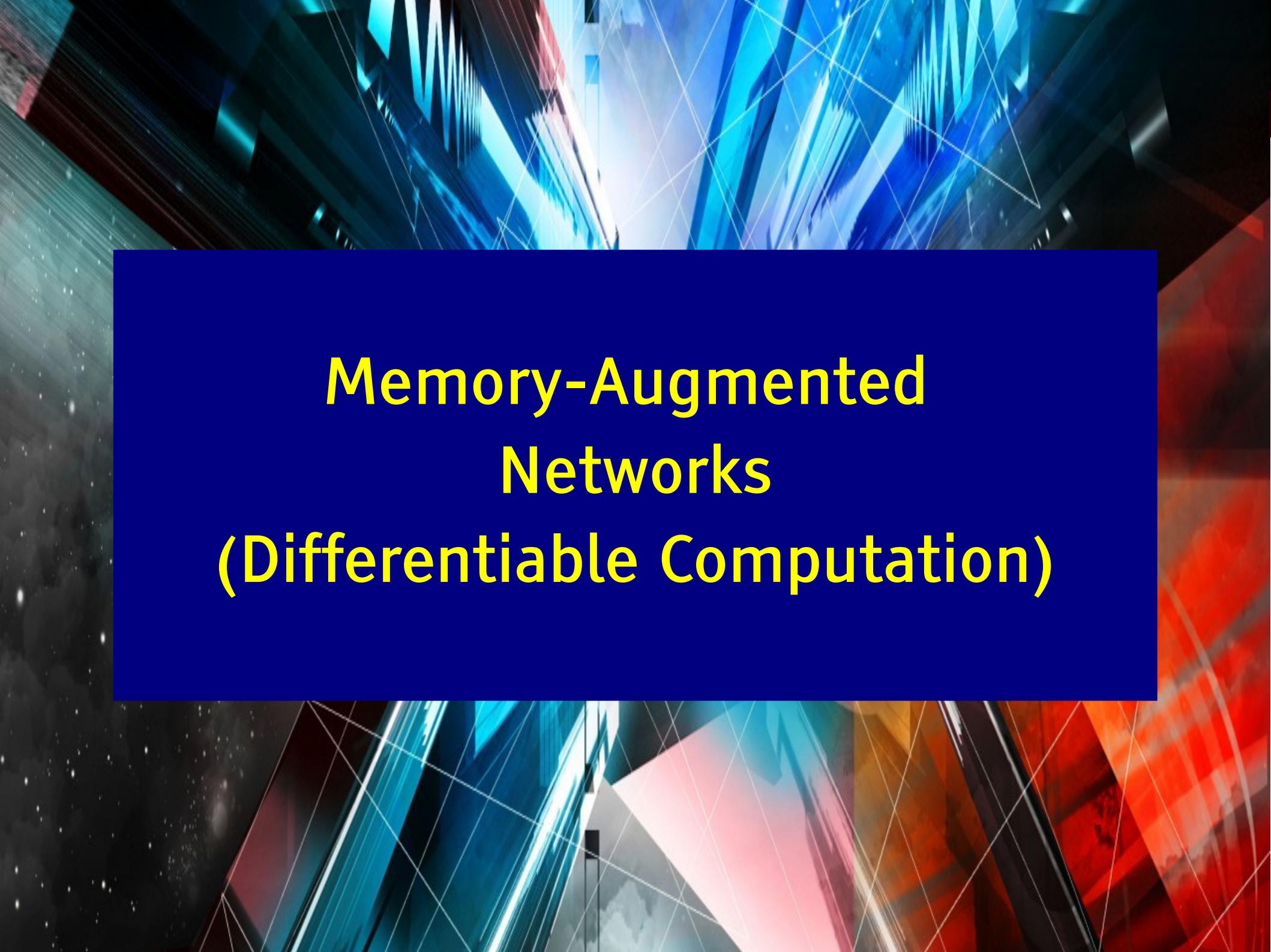
- Compute graph Laplacian
- Compute transformation into its eigenspace
 - ▶ Fourier transform
- Learn “smooth” pointwise multipliers in Fourier space
 - ▶ Localized kernels
- Applicable to any function on graphs
 - ▶ Chemistry, text, images with holes, image on non-euclidean surfaces...
 - ▶ [Bruna et al.
Arxiv:1312.6203]
 - ▶ [Henaff et al.
Arxiv:1506.05163]



(c)

(d)





Memory-Augmented Networks (Differentiable Computation)

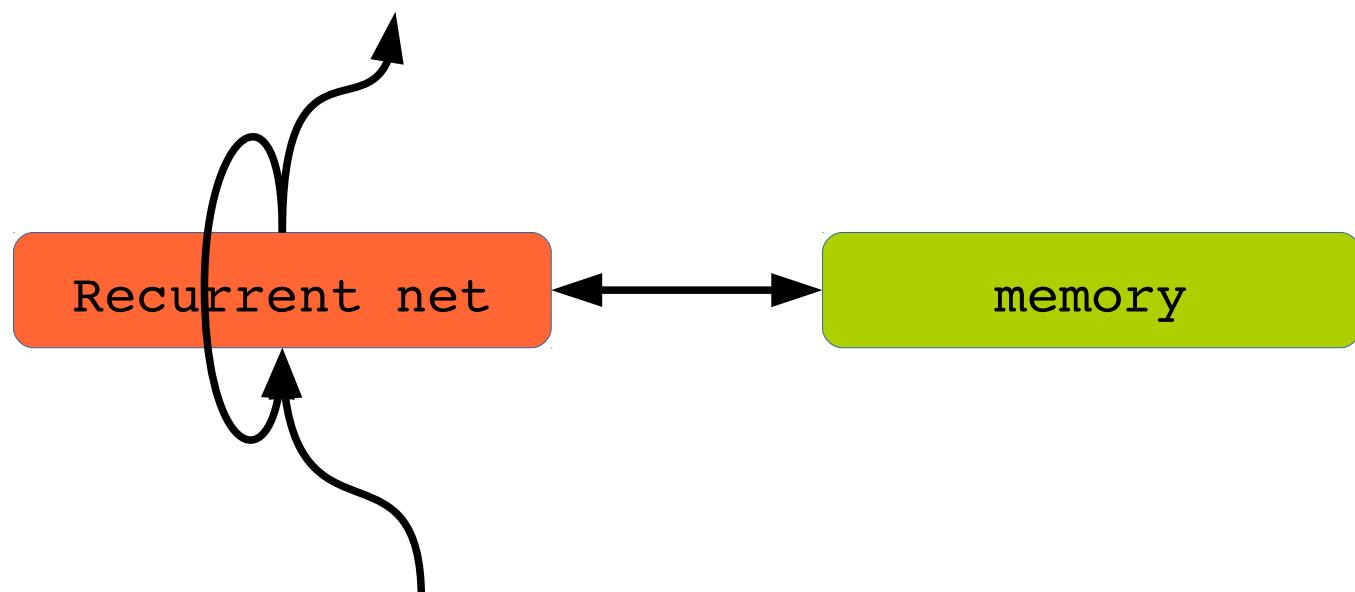
f Augmenting Neural Nets with a Memory Module

▀ 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
- ▶ **Stacked-Augmented Recurrent Neural Net** [Joulin & Mikolov 2014] (FAIR)
- ▶ NTM [DeepMind 2014], “tape”.



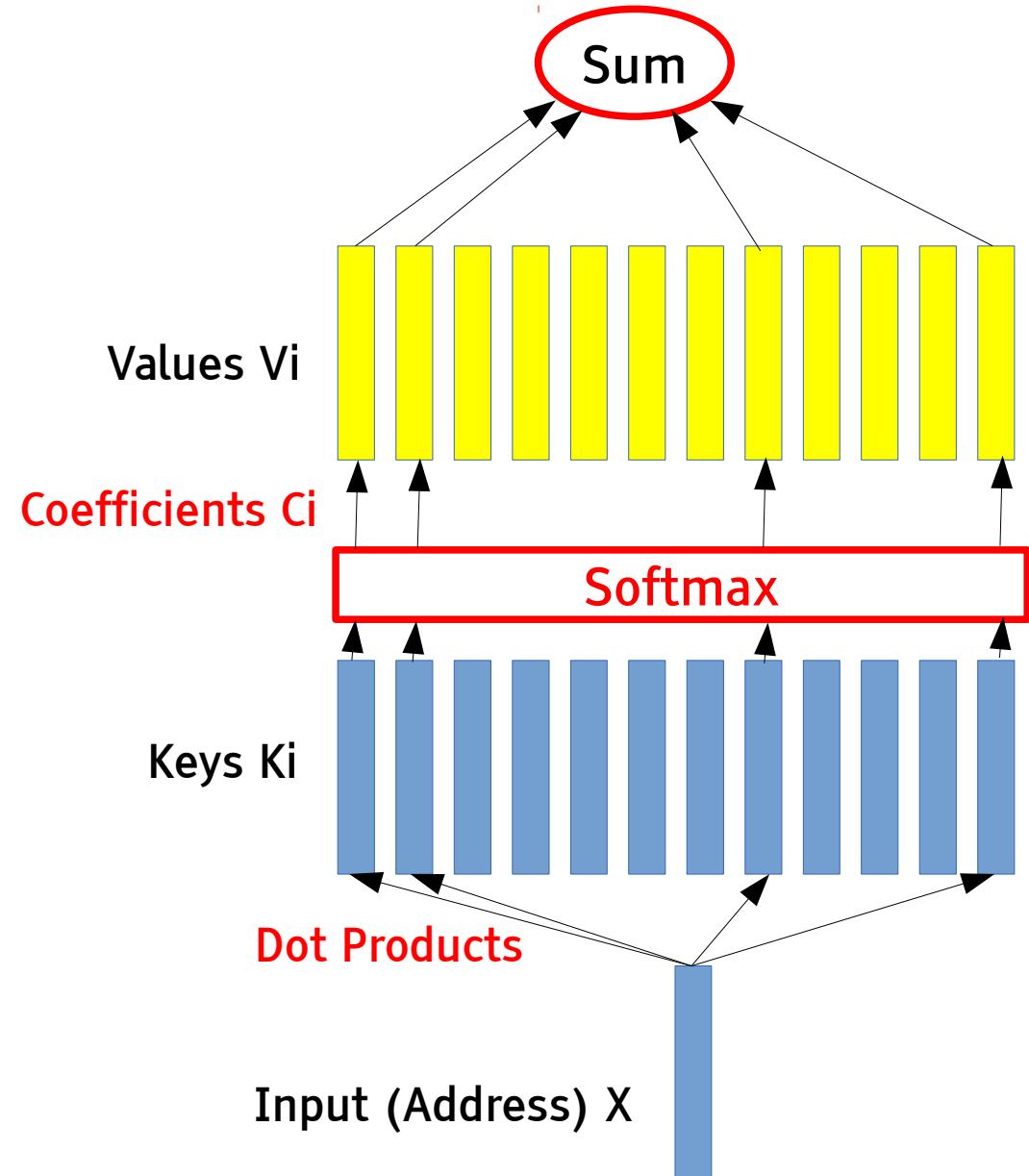
Differentiable Memory

Y LeCun

- Like a “soft” RAM circuit
- Or a “soft” hash table
- Stores Key-Value pairs (K_i, V_i)

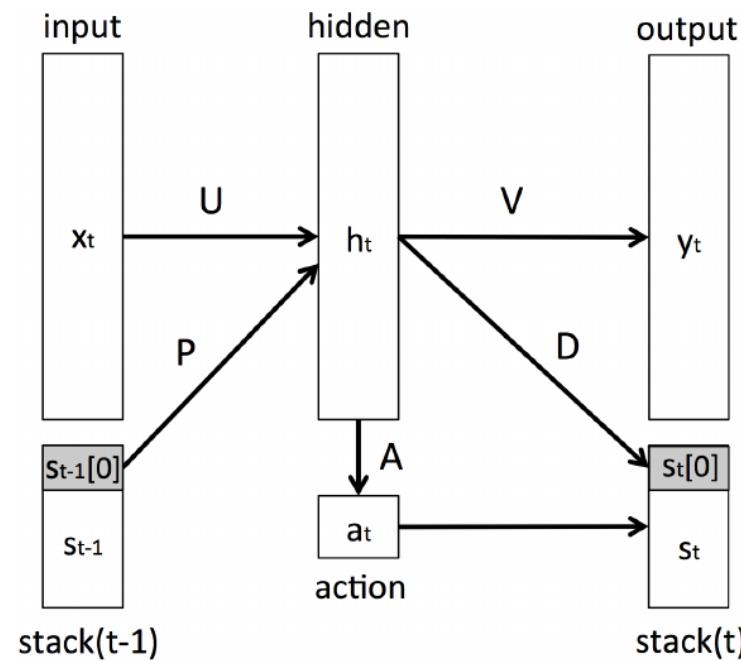
$$C_i = \frac{e^{K_i^T X}}{\sum_j e^{K_j^T X}}$$

$$Y = \sum_i C_i V_i$$

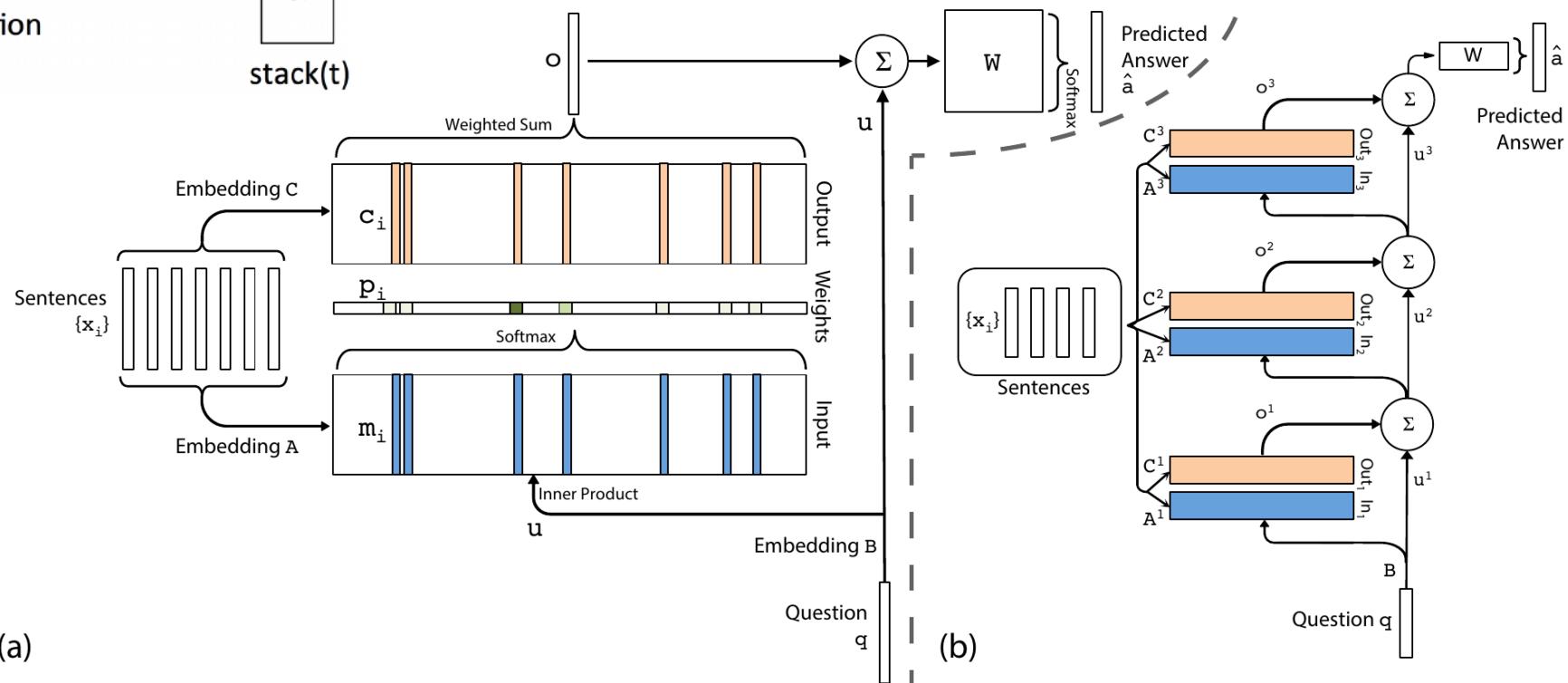


Memory/Stack-Augmented Recurrent Nets

Y LeCun



- [Joulin & Mikolov, ArXiv:1503.01007]
 - ▶ Stack-augmented RNN
- [Sukhbataar, Szlam, Weston, Fergus NIPS 2015]
 - ▶ ArXiv:1503.08895]
- Weakly-supervised MemNN:
 - ▶ discovers which memory location to use.





Memory Network [Weston, Chopra, Bordes 2014]

Add a short-term memory to a network

<http://arxiv.org/abs/1410.39>

16

- 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.

```

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

```

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

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.

End-to-End Memory Network on bAbI tasks [Weston 2015]

Y LeCun

Sam walks into the kitchen.
 Sam picks up an apple.
 Sam walks into the bedroom.
 Sam drops the apple.
Q: Where is the apple?
A. Bedroom

Brian is a lion.
 Julius is a lion.
 Julius is white.
 Bernhard is green.
Q: What color is Brian?
A. White

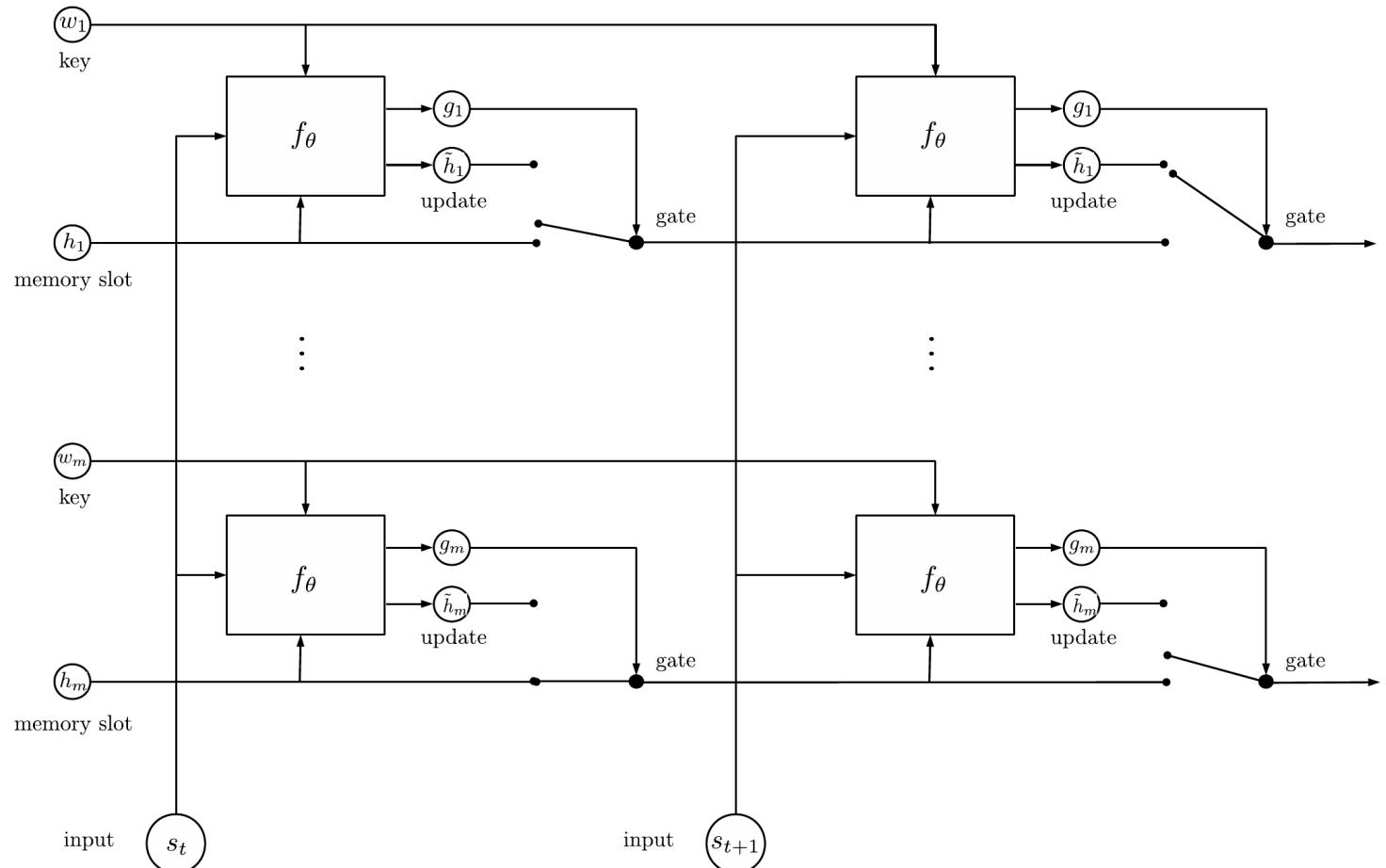
Mary journeyed to the den.
 Mary went back to the kitchen.
 John journeyed to the bedroom.
 Mary discarded the milk.
Q: Where was the milk before the den?
A. Hallway

Task	Baseline			MemN2N								
	Strongly Supervised MemNN [21]	LSTM [21]	MemNN WSH	BoW	PE	PE LS	PE RN	1 hop PE LS joint	2 hops PE LS joint	3 hops PE LS joint	PE LS RN joint	PE LW joint
1: 1 supporting fact	0.0	50.0	0.1	0.6	0.1	0.2	0.0	0.8	0.0	0.1	0.0	0.1
2: 2 supporting facts	0.0	80.0	42.8	17.6	21.6	12.8	8.3	62.0	15.6	14.0	11.4	18.8
3: 3 supporting facts	0.0	80.0	76.4	71.0	64.2	58.8	40.3	76.9	31.6	33.1	21.9	31.7
4: 2 argument relations	0.0	39.0	40.3	32.0	3.8	11.6	2.8	22.8	2.2	5.7	13.4	17.5
5: 3 argument relations	2.0	30.0	16.3	18.3	14.1	15.7	13.1	11.0	13.4	14.8	14.4	12.9
6: yes/no questions	0.0	52.0	51.0	8.7	7.9	8.7	7.6	7.2	2.3	3.3	2.8	2.0
7: counting	15.0	51.0	36.1	23.5	21.6	20.3	17.3	15.9	25.4	17.9	18.3	10.1
8: lists/sets	9.0	55.0	37.8	11.4	12.6	12.7	10.0	13.2	11.7	10.1	9.3	6.1
9: simple negation	0.0	36.0	35.9	21.1	23.3	17.0	13.2	5.1	2.0	3.1	1.9	1.5
10: indefinite knowledge	2.0	56.0	68.7	22.8	17.4	18.6	15.1	10.6	5.0	6.6	6.5	2.6
11: basic coreference	0.0	38.0	30.0	4.1	4.3	0.0	0.9	8.4	1.2	0.9	0.3	3.3
12: conjunction	0.0	26.0	10.1	0.3	0.3	0.1	0.2	0.4	0.0	0.3	0.1	0.0
13: compound coreference	0.0	6.0	19.7	10.5	9.9	0.3	0.4	6.3	0.2	1.4	0.2	0.5
14: time reasoning	1.0	73.0	18.3	1.3	1.8	2.0	1.7	36.9	8.1	8.2	6.9	2.0
15: basic deduction	0.0	79.0	64.8	24.3	0.0	0.0	0.0	46.4	0.5	0.0	0.0	1.8
16: basic induction	0.0	77.0	50.5	52.0	52.1	1.6	1.3	47.4	51.3	3.5	2.7	51.0
17: positional reasoning	35.0	49.0	50.9	45.4	50.1	49.0	51.0	44.4	41.2	44.5	40.4	42.6
18: size reasoning	5.0	48.0	51.3	48.1	13.6	10.1	11.1	9.6	10.3	9.2	9.4	9.2
19: path finding	64.0	92.0	100.0	89.7	87.4	85.6	82.8	90.7	89.9	90.2	88.0	90.6
20: agent's motivation	0.0	9.0	3.6	0.1	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.2
Mean error (%)	6.7	51.3	40.2	25.1	20.3	16.3	13.9	25.8	15.6	13.3	12.4	15.2
Failed tasks (err. > 5%)	4	20	18	15	13	12	11	17	11	11	11	10
On 10k training data												
Mean error (%)	3.2	36.4	39.2	15.4	9.4	7.2	6.6	24.5	10.9	7.9	7.5	11.0
Failed tasks (err. > 5%)	2	16	17	9	6	4	4	16	7	6	6	6

Entity Recurrent Neural Net

Y LeCun

- Maintains a current estimate of the state of the world.
- Each module is a recurrent net with a “memory”
- Each input event causes some of the memory cells to get updated



EntNet is the first model to solve all 20 bAbI tasks

Y LeCun

Task	D-NTM	MemN2N	DNC	DMN+	EntNet
1: 1 supporting fact	4.4	0	0	0	0
2: 2 supporting facts	27.5	0.3	0.4	0.3	0.1
3: 3 supporting facts	71.3	2.1	1.8	1.1	4.1
4: 2 argument relations	0	0	0	0	0
5: 3 argument relations	1.7	0.8	0.8	0.5	0.3
6: yes/no questions	1.5	0.1	0	0	0.2
7: counting	6.0	2.0	0.6	2.4	0
8: lists/sets	1.7	0.9	0.3	0.0	0.5
9: simple negation	0.6	0.3	0.2	0.0	0.1
10: indefinite knowledge	19.8	0	0.2	0	0.6
11: basic coreference	0	0.0	0	0.0	0.3
12: conjunction	6.2	0	0	0.2	0
13: compound coreference	7.5	0	0	0	1.3
14: time reasoning	17.5	0.2	0.4	0.2	0
15: basic deduction	0	0	0	0	0
16: basic induction	49.6	51.8	55.1	45.3	0.2
17: positional reasoning	1.2	18.6	12.0	4.2	0.5
18: size reasoning	0.2	5.3	0.8	2.1	0.3
19: path finding	39.5	2.3	3.9	0.0	2.3
20: agent's motivation	0	0	0	0	0
Failed Tasks (> 5% error):	9	3	2	1	0
Mean Error:	12.8	4.2	3.8	2.8	0.5



EntNet is the first model to solve all 20 bAbI tasks

Y LeCun

Story	Key	1-NN	2-NN
mary got the milk there john moved to the bedroom sandra went back to the kitchen mary travelled to the hallway john got the football there john went to the hallway john put down the football mary went to the garden john went to the kitchen sandra travelled to the hallway daniel went to the hallway mary discarded the milk where is the milk ? answer: garden	football milk john mary sandra daniel bedroom kitchen garden hallway	hallway (0.135) garden (0.111) kitchen (0.501) garden (0.442) hallway (0.394) hallway (0.689) hallway (0.367) kitchen (0.483) garden (0.281) hallway (0.475)	dropped (0.056) took (0.011) dropped (0.027) took (0.034) kitchen (0.121) to (0.076) dropped (0.075) daniel (0.029) where (0.026) left (0.060)

Obstacles to AI



Obstacles to Progress in AI

Machines need to learn/understand how the world works

- ▶ Physical world, digital world, people,....
- ▶ They need to acquire some level of common sense

They need to learn a very large amount of background knowledge

- ▶ Through observation and action

Machines need to perceive the state of the world

- ▶ So as to make accurate predictions and planning

Machines need to update and remember estimates of the state of the world

- ▶ Paying attention to important events. Remember relevant events

Machines need to reason and plan

- ▶ Predict which sequence of actions will lead to a desired state of the world

Intelligence & Common Sense =

Perception + Predictive Model + Memory + Reasoning & Planning

What is Common Sense?

Y LeCun

- “The trophy doesn’t fit in the suitcase because it’s too large/small”
- “Tom picked up his bag and left the room”
- We have common sense because we know how the world works
- How do we get machines to learn that?



How Much Information Does the Machine Need to Predict?

Y LeCun

■ Reinforcement Learning (cherry)

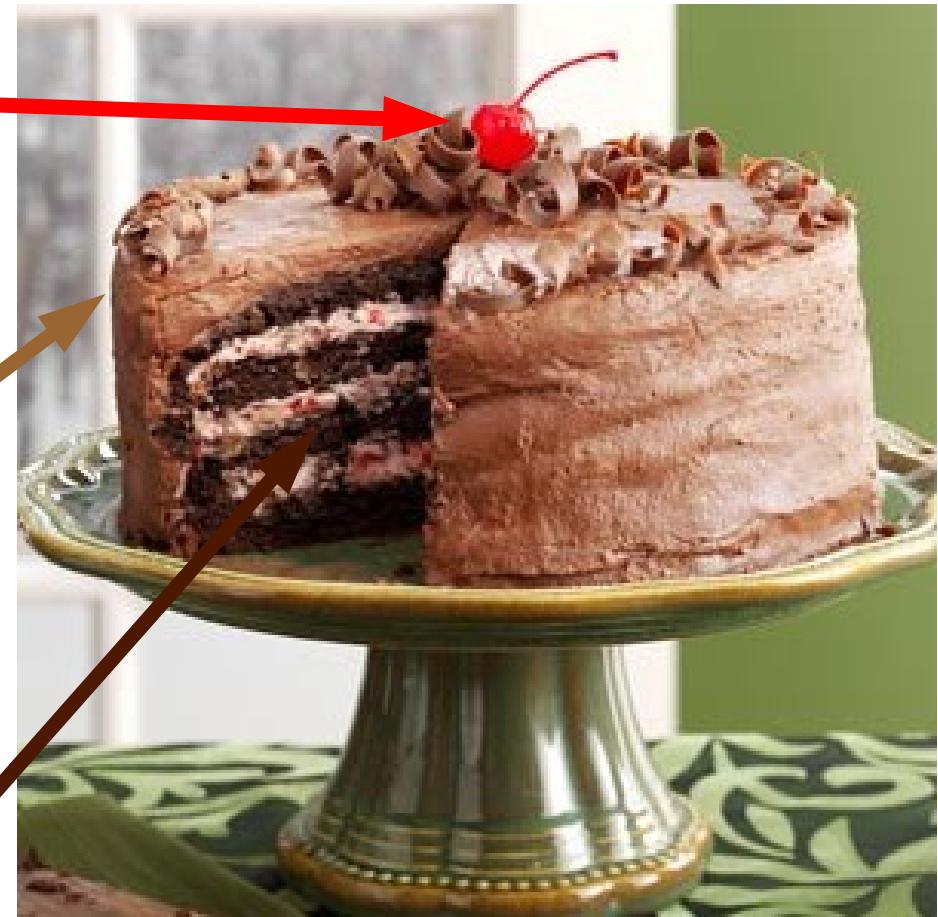
- ▶ The machine predicts a scalar reward given once in a while.
- ▶ **A few bits for some samples**

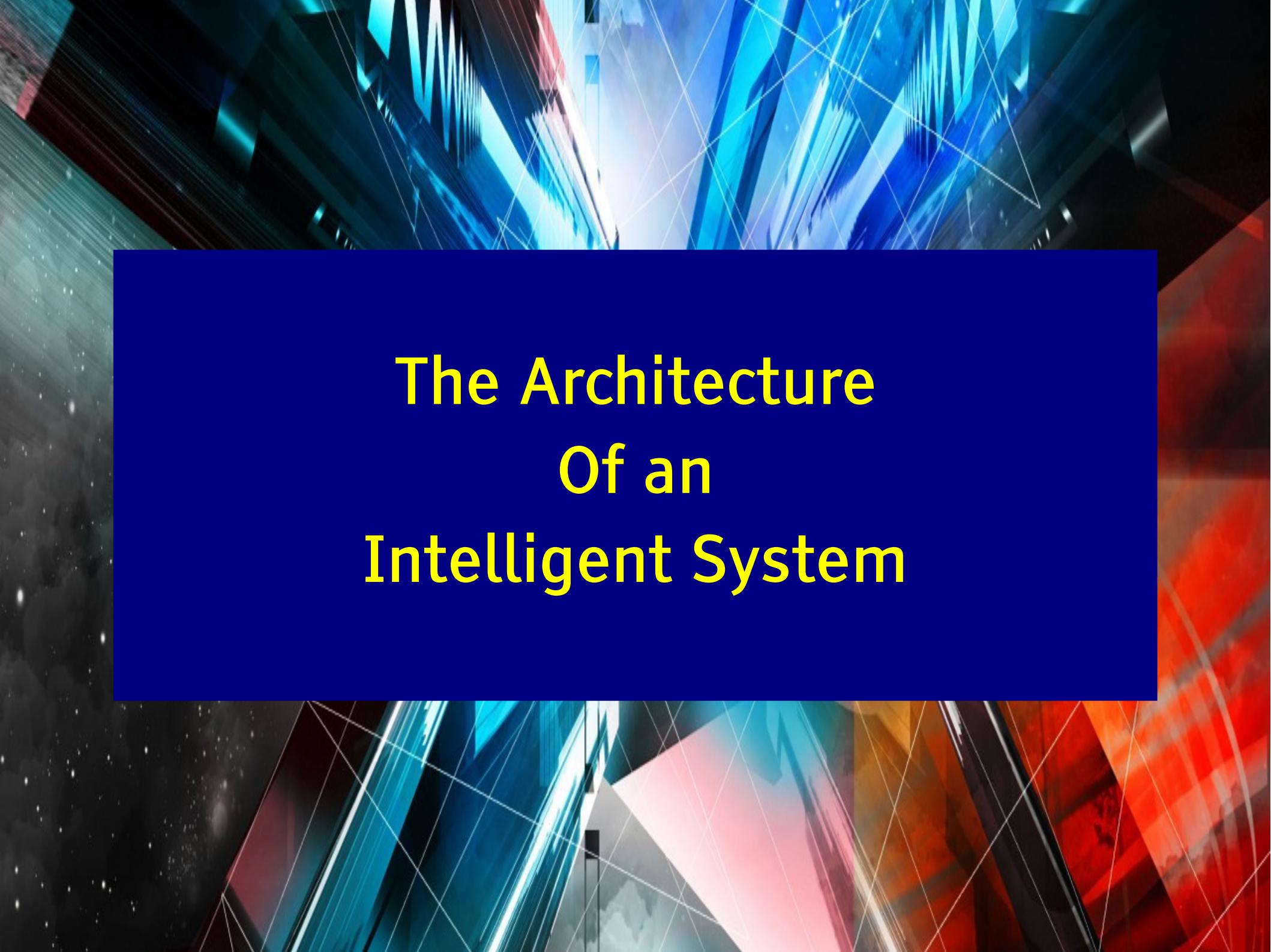
■ Supervised Learning (icing)

- ▶ The machine predicts a category or a few numbers for each input
- ▶ **10→10,000 bits per sample**

■ Unsupervised Learning (cake)

- ▶ The machine predicts any part of its input for any observed part.
- ▶ Predicts future frames in videos
- ▶ **Millions of bits per sample**



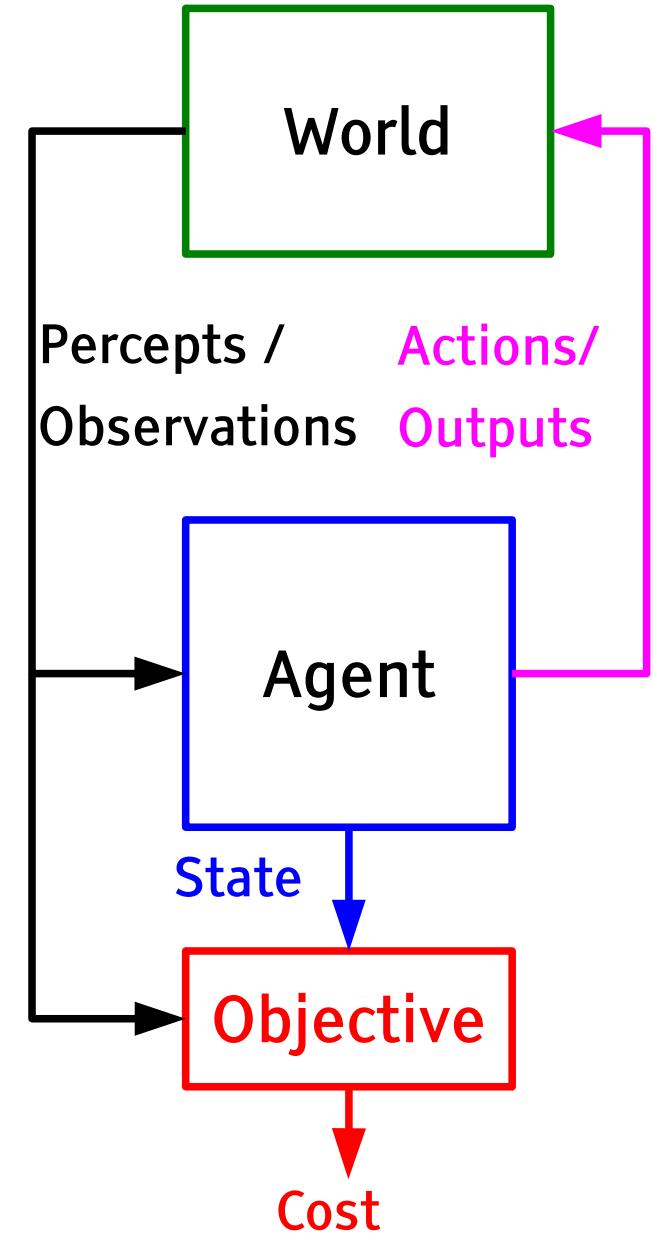
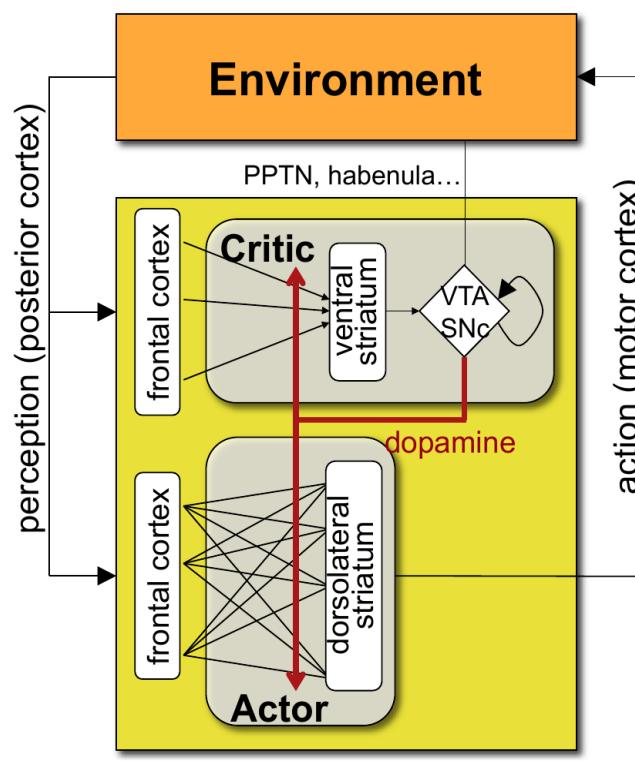
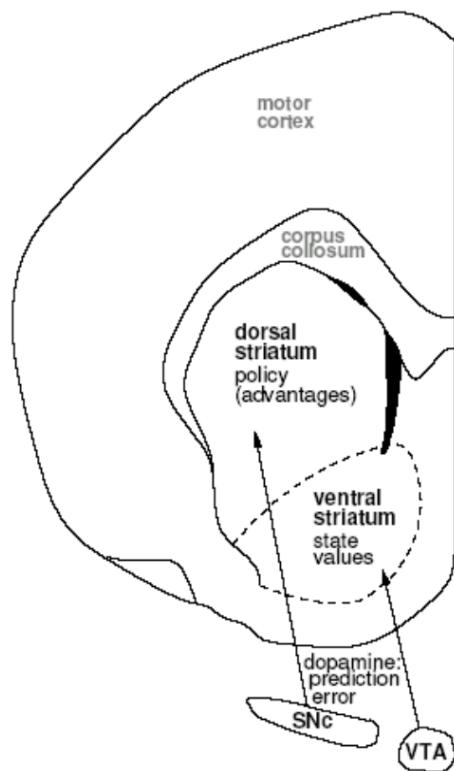


The Architecture Of an Intelligent System



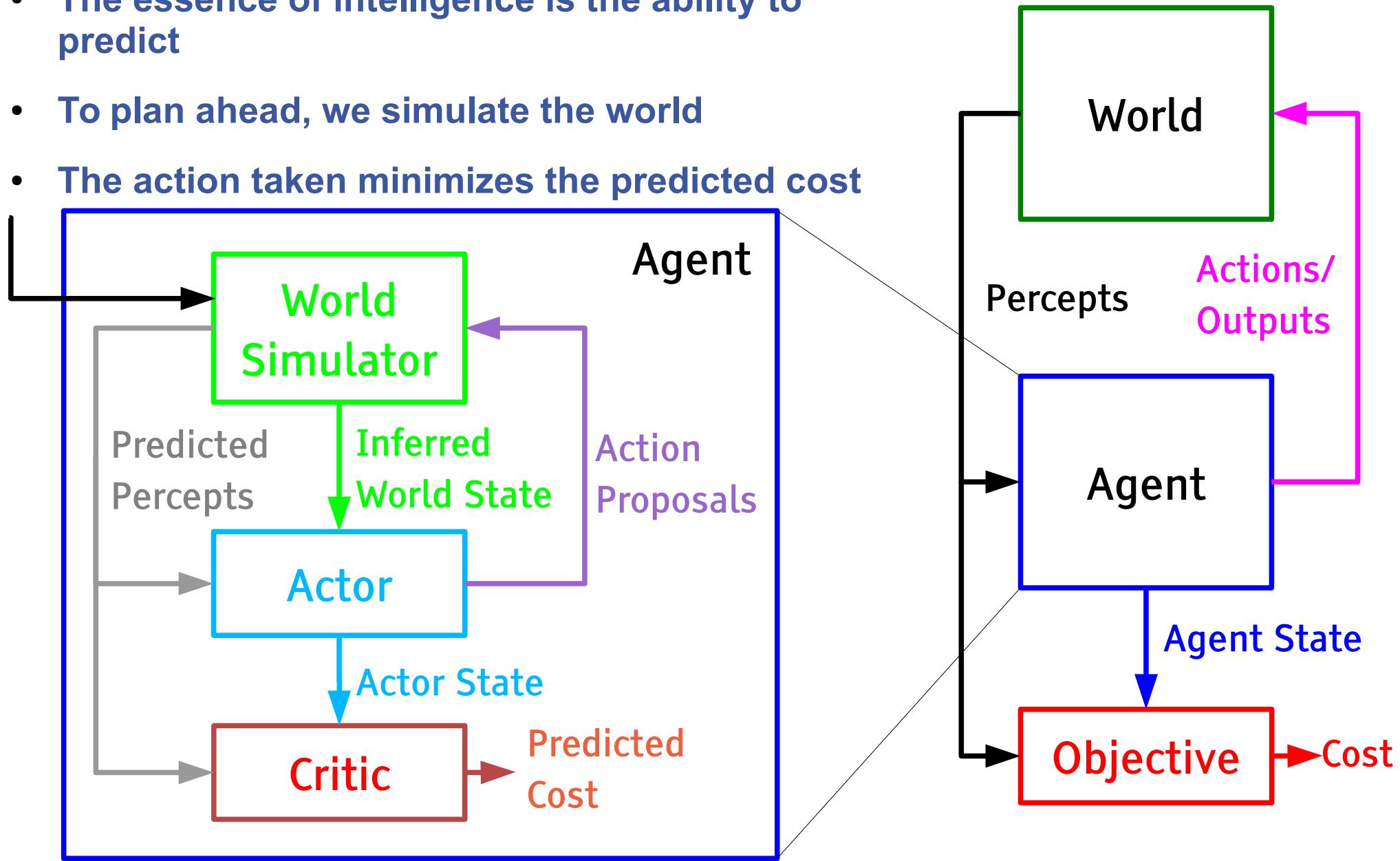
AI System: Learning Agent + Immutable Objective

- The agent gets percepts from the world
- The agent acts on the world
- The agents tries to minimize the long-term expected cost.



AI System: Predicting + Planning = Reasoning

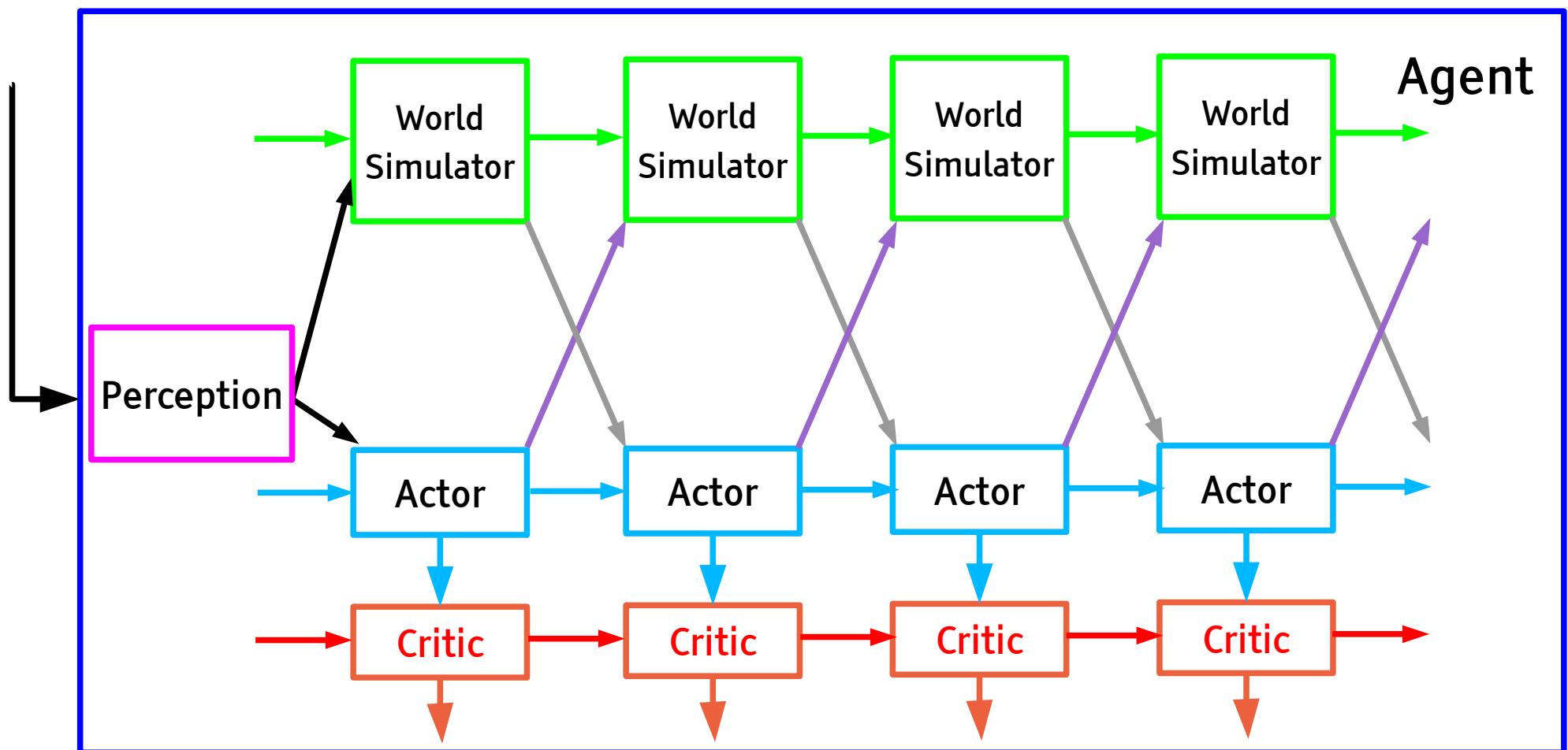
- The essence of intelligence is the ability to predict
- To plan ahead, we simulate the world
- The action taken minimizes the predicted cost

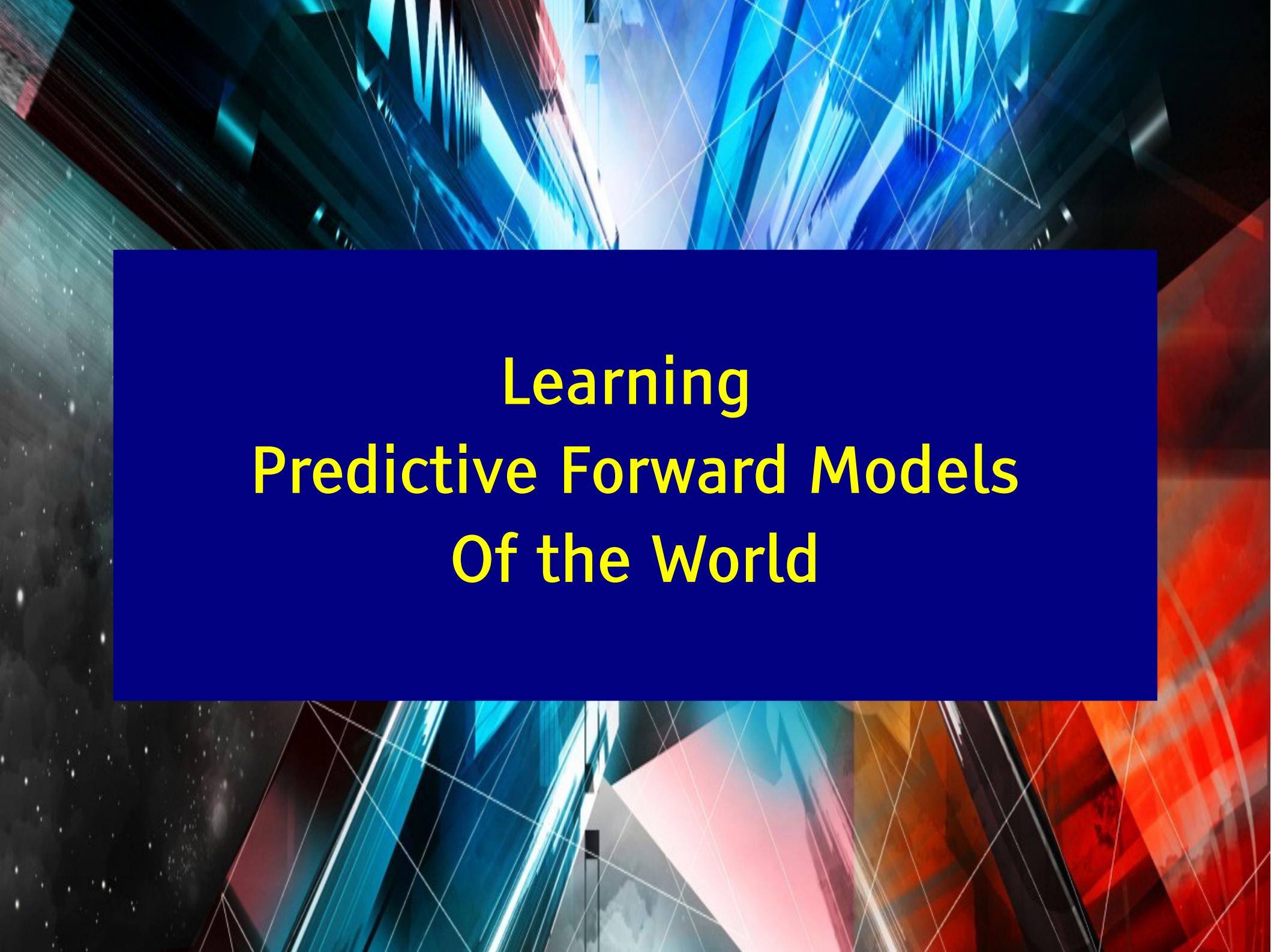




AI System: Predicting + Planning = Reasoning

- The essence of intelligence is the ability to predict
- To plan ahead, we must simulate the world
- The action taken minimizes the predicted cost





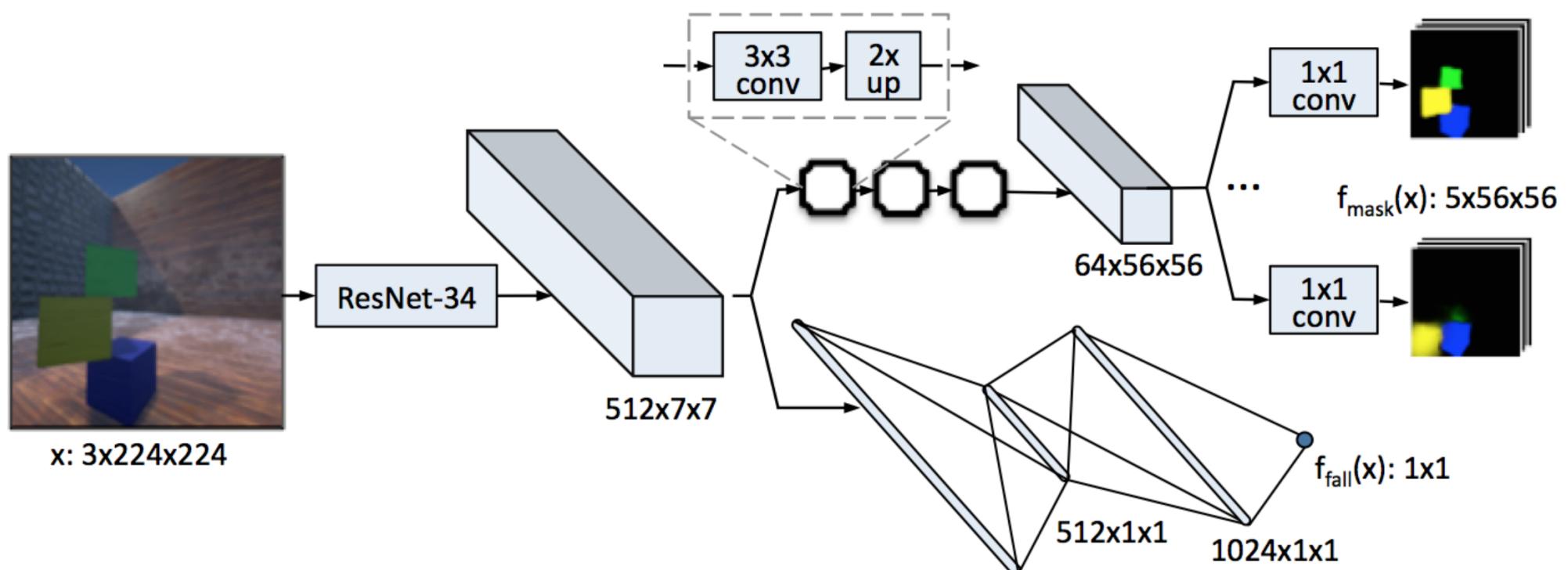
Learning Predictive Forward Models Of the World

Learning Physics (PhysNet)

Y LeCun

[Lerer, Gross, Fergus arxiv:1603.01312]

- ▶ ConvNet produces object masks that predict the trajectories of falling blocks
- ▶ Uses the Unreal game engine.



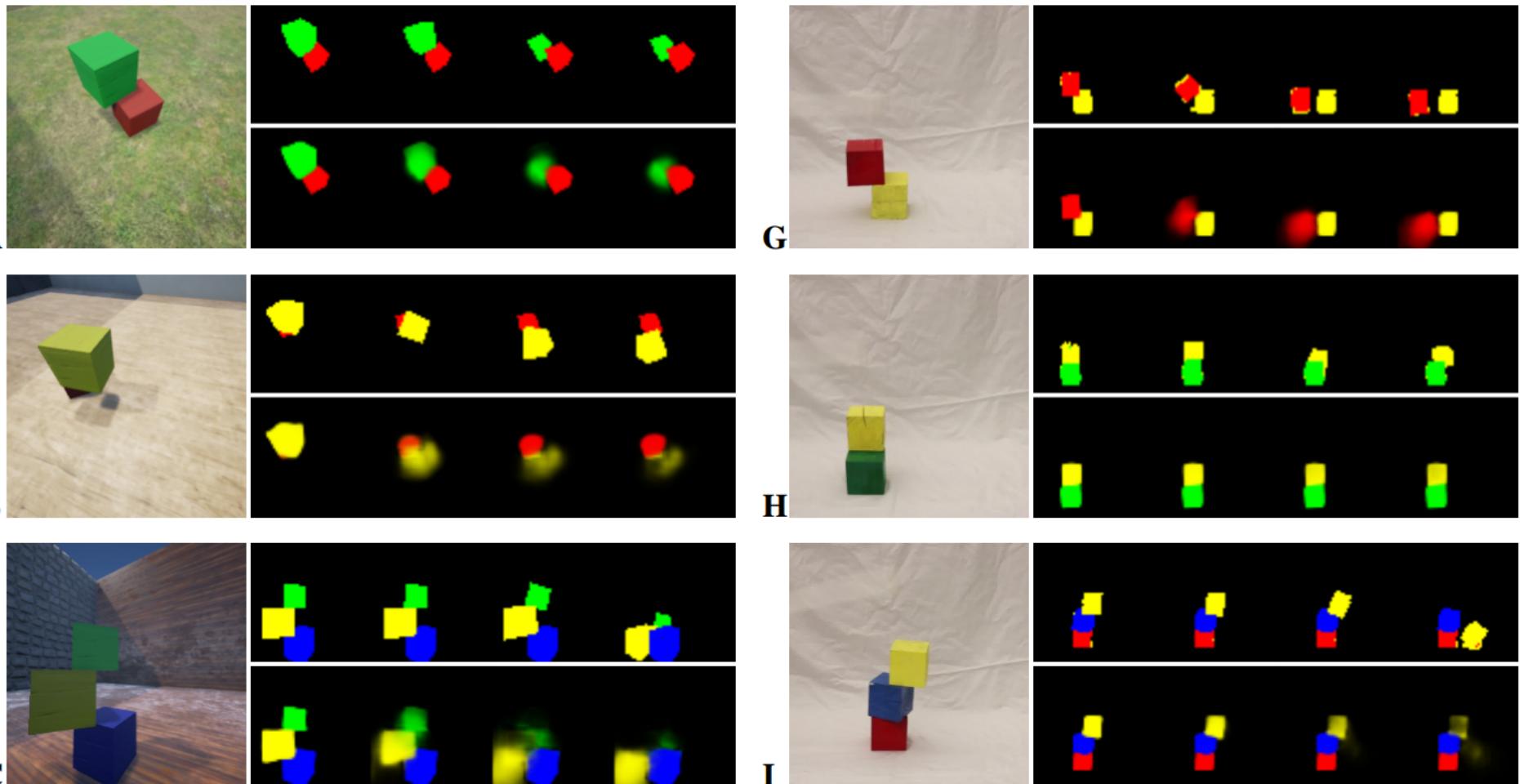


Learning Physics (PhysNet)

Y LeCun

[Lerer, Gross, Fergus arxiv:1603.01312]

- ▶ ConvNet produces object masks that predict the trajectories of falling blocks
- ▶ Uses the Unreal game engine.



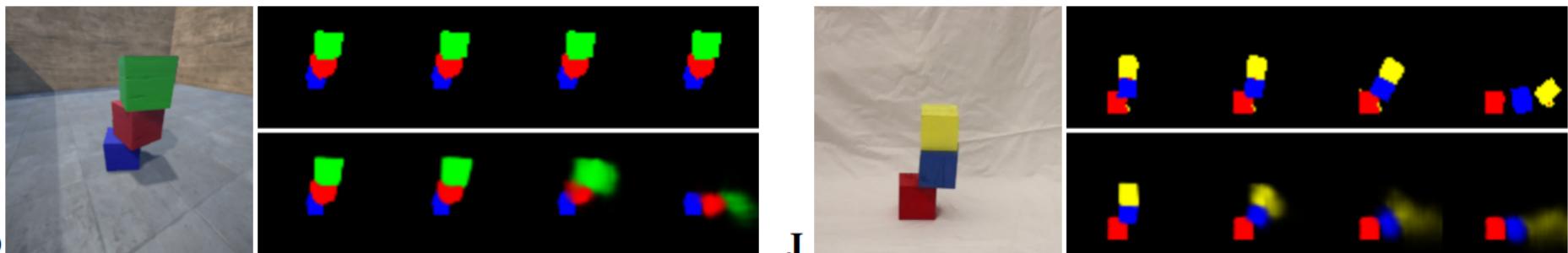


Learning Physics (PhysNet)

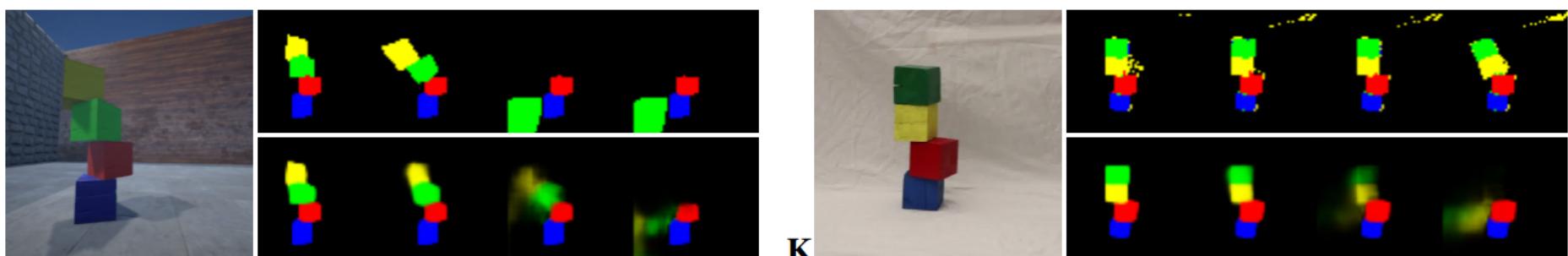
Y LeCun

[Lerer, Gross, Fergus arxiv:1603.01312]

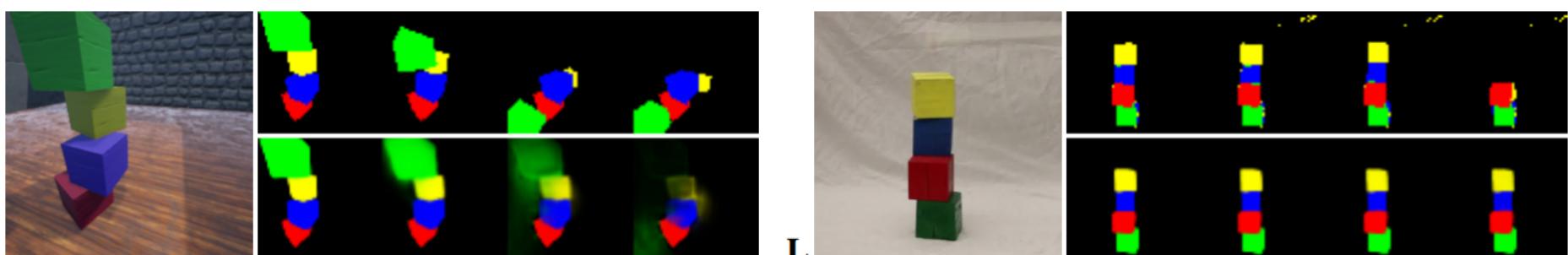
- ▶ ConvNet produces object masks that predict the trajectories of falling blocks
- ▶ Uses the Unreal game engine.



J



K



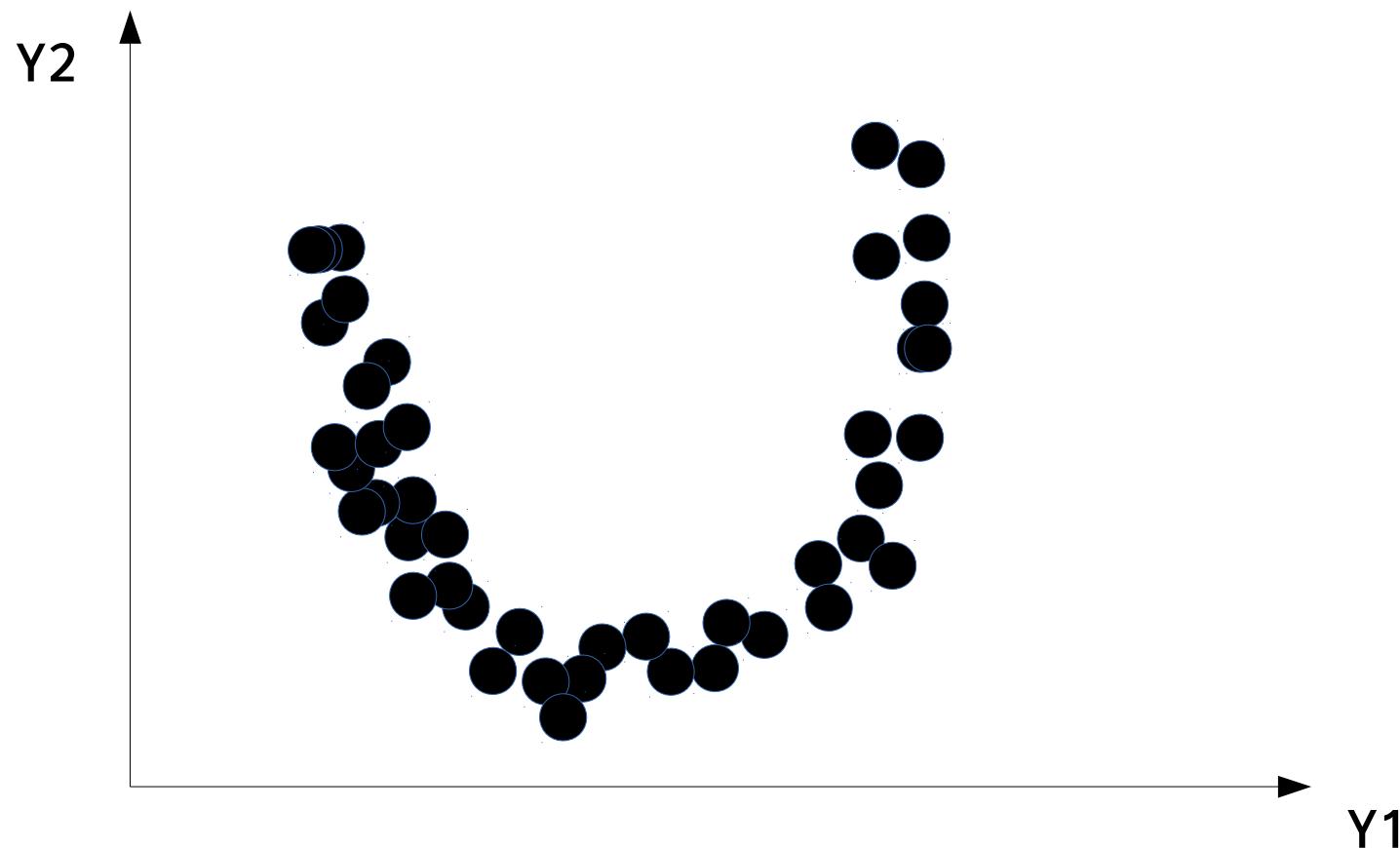
L

Unsupervised Learning

Energy-Based Unsupervised Learning

Y LeCun

- Learning an **energy function** (or contrast function) that takes
 - ▶ Low values on the data manifold
 - ▶ Higher values everywhere else

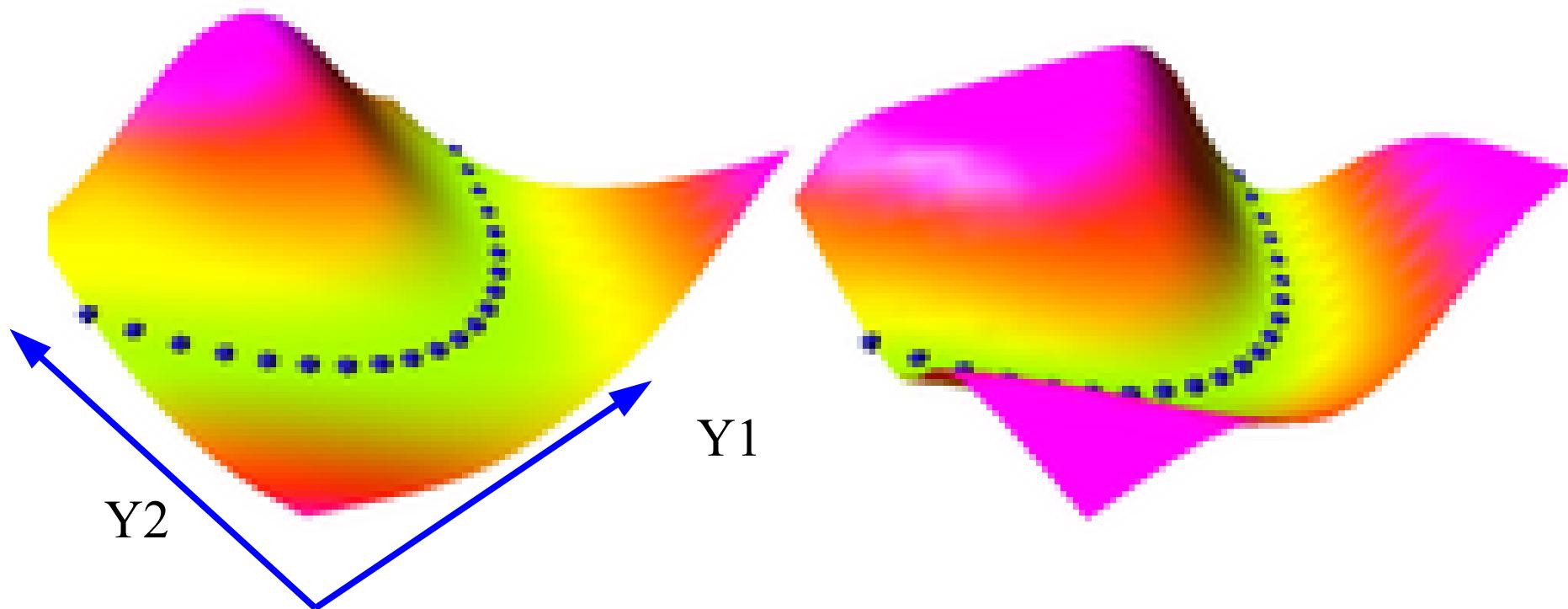


Capturing Dependencies Between Variables with an Energy Function

Y LeCun

- The energy surface is a “contrast function” that takes low values on the data manifold, and higher values everywhere else
 - Special case: energy = negative log density
 - Example: the samples live in the manifold

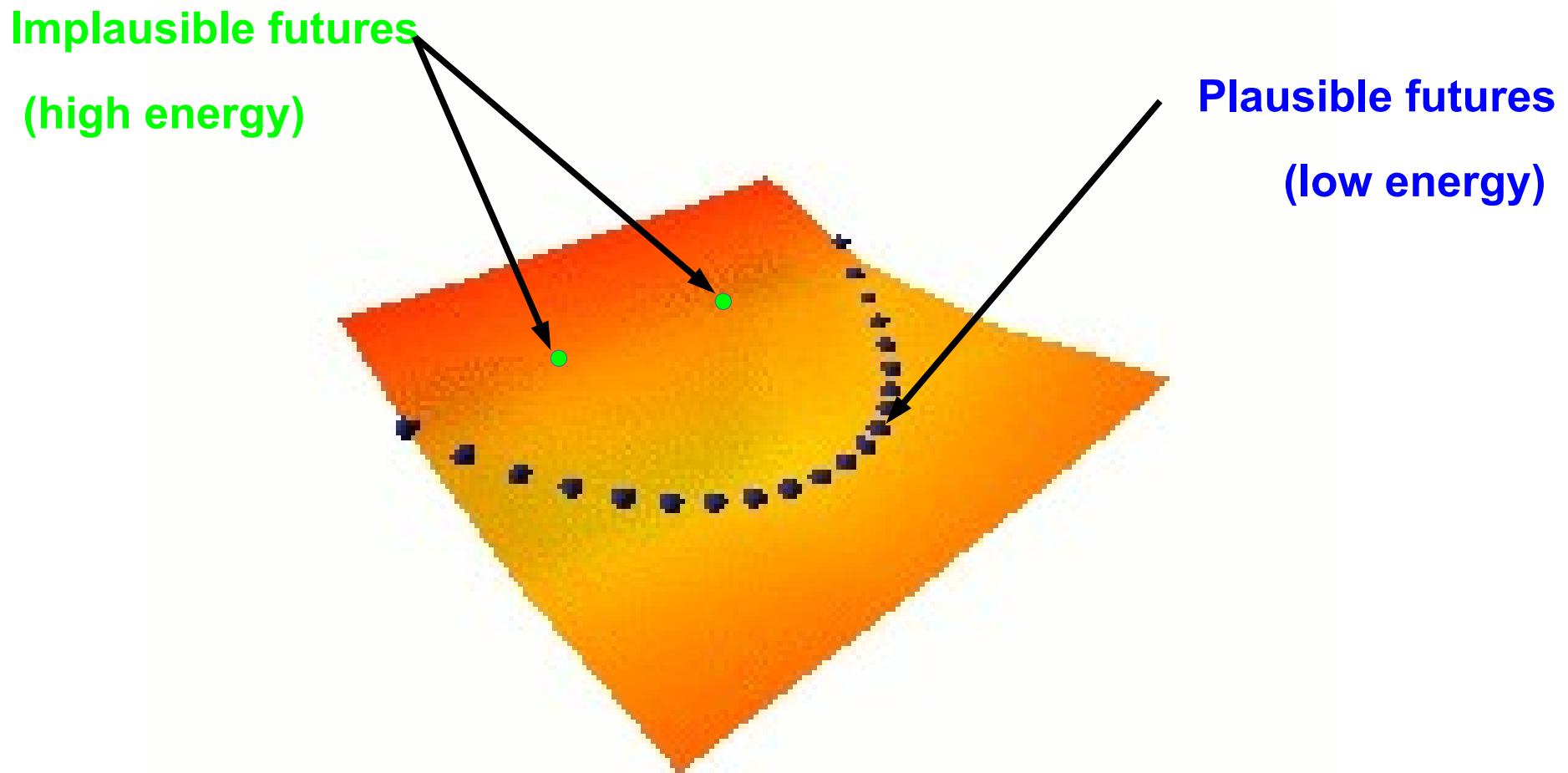
$$Y_2 = (Y_1)^2$$





Energy-Based Unsupervised Learning

- Energy Function: Takes low value on data manifold, higher values everywhere else
- Push down on the energy of desired outputs. Push up on everything else.
- **But how do we choose where to push up?**

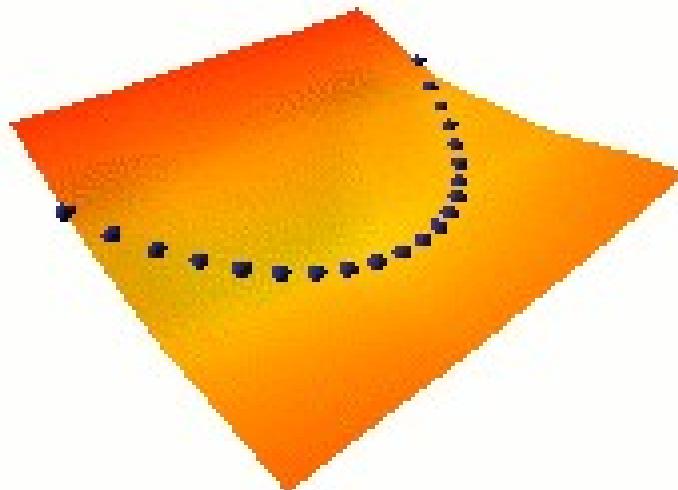
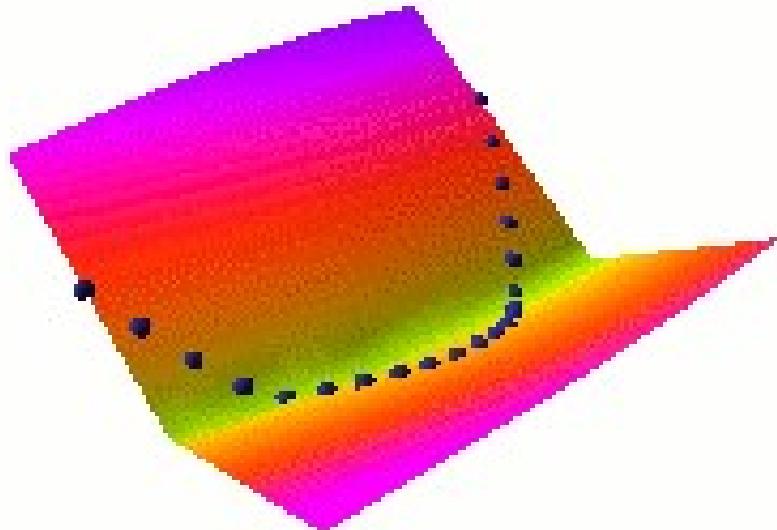


Learning the Energy Function

Y LeCun

parameterized energy function $E(Y,W)$

- ▶ Make the energy low on the samples
- ▶ Make the energy higher everywhere else
- ▶ Making the energy low on the samples is easy
- ▶ But how do we make it higher everywhere else?





Seven Strategies to Shape the Energy Function

Y LeCun

- 1. build the machine so that the volume of low energy stuff is constant
 - ▶ PCA, K-means, GMM, square ICA
- 2. push down of the energy of data points, push up everywhere else
 - ▶ Max likelihood (needs tractable partition function)
- 3. push down of the energy of data points, push up on chosen locations
 - ▶ contrastive divergence, Ratio Matching, Noise Contrastive Estimation, Minimum Probability Flow
- 4. minimize the gradient and maximize the curvature around data points
 - ▶ score matching
- 5. train a dynamical system so that the dynamics goes to the manifold
 - ▶ denoising auto-encoder
- 6. use a regularizer that limits the volume of space that has low energy
 - ▶ Sparse coding, sparse auto-encoder, PSD
- 7. if $E(Y) = \|Y - G(Y)\|^2$, make $G(Y)$ as "constant" as possible.
 - ▶ Contracting auto-encoder, saturating auto-encoder

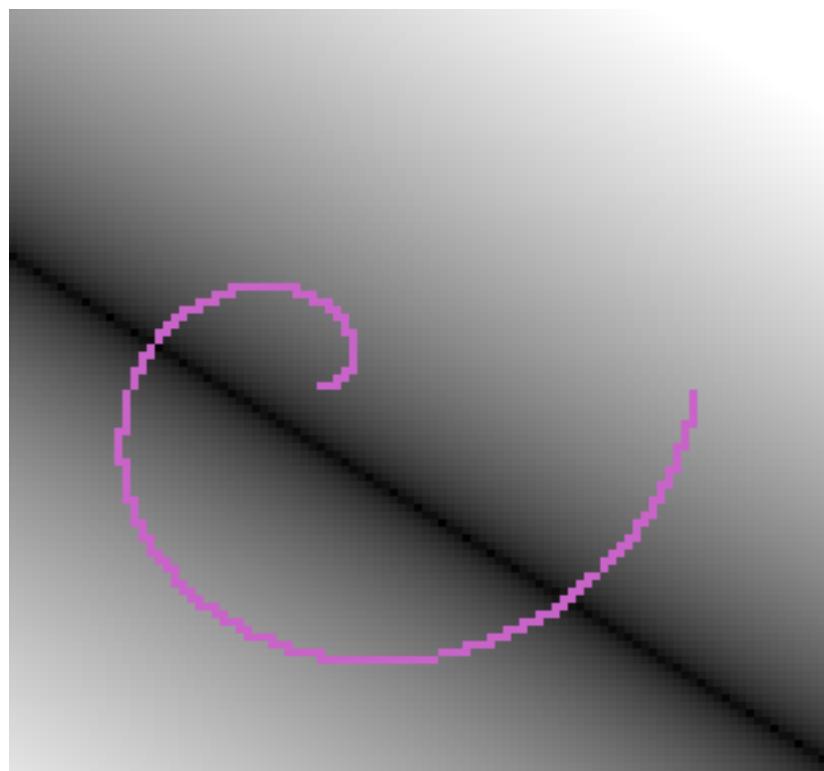
#1: constant volume of low energy Energy surface for PCA and K-means

Y LeCun

- 1. build the machine so that the volume of low energy stuff is constant
 - ▶ PCA, K-means, GMM, square ICA...

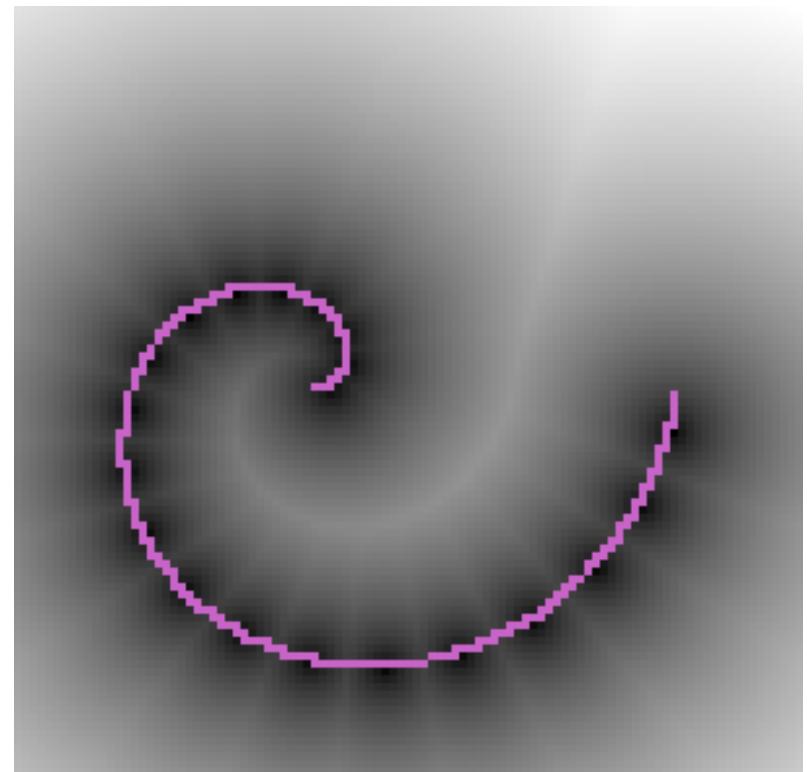
PCA

$$E(Y) = \|W^T W Y - Y\|^2$$



K-Means,
Z constrained to 1-of-K code

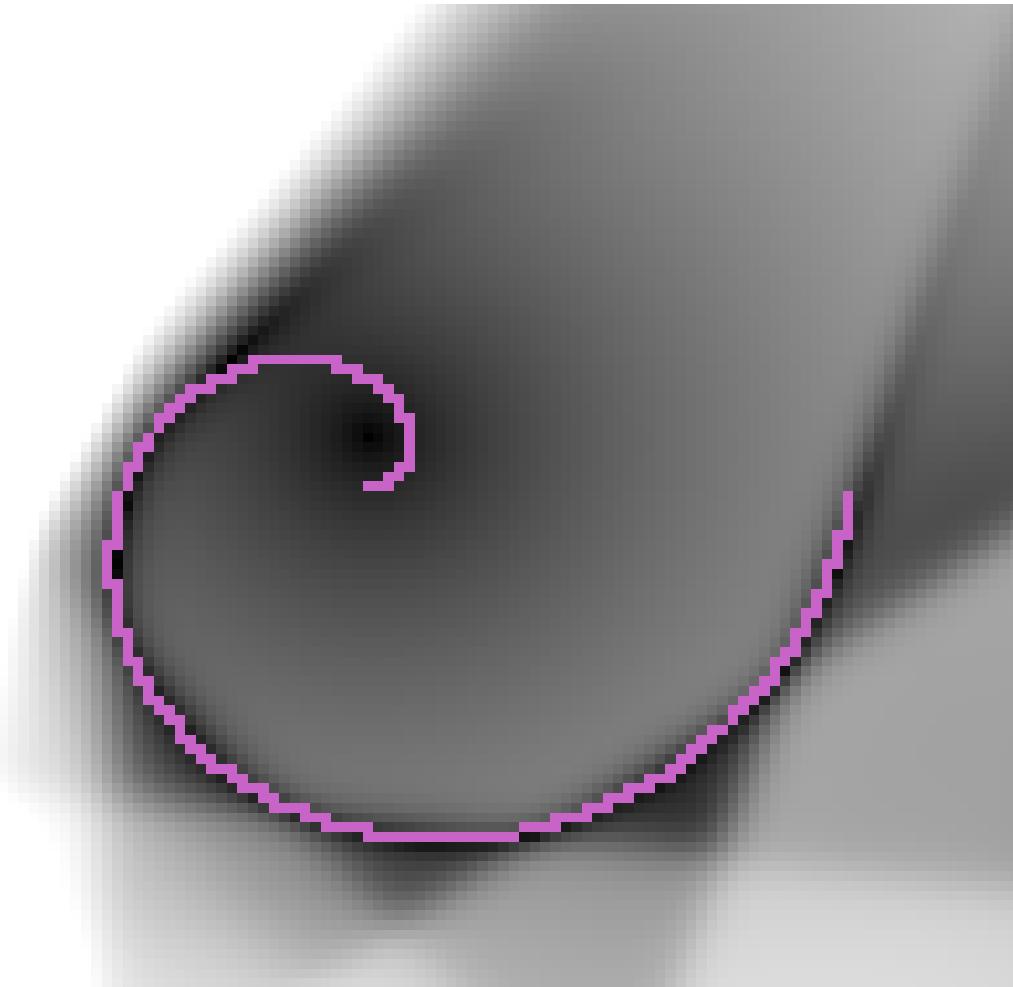
$$E(Y) = \min_z \sum_i \|Y - W_i Z_i\|^2$$



#6. use a regularizer that limits
the volume of space that has low energy

Y LeCun

Sparse coding, sparse auto-encoder, Predictive Sparse Decomposition

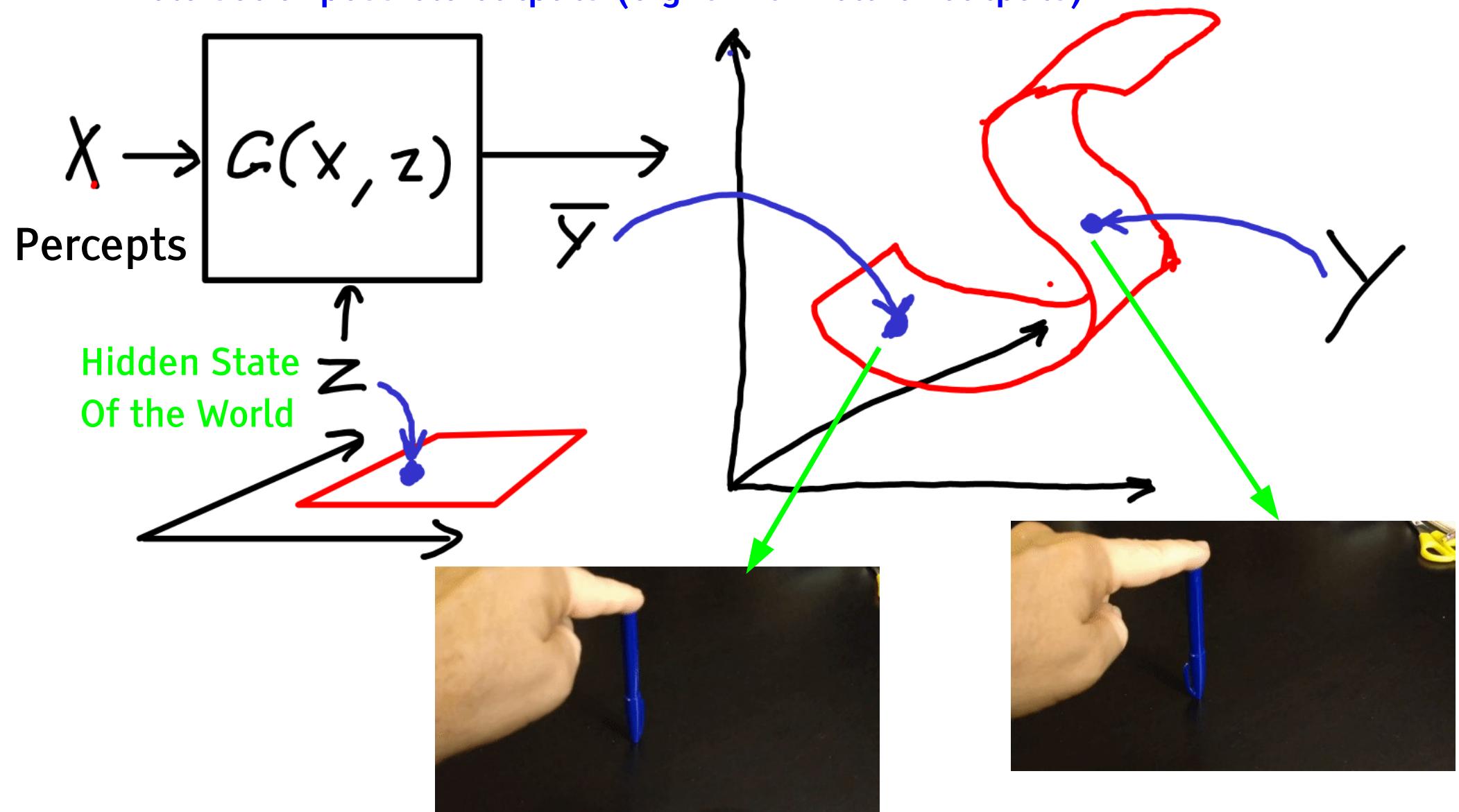


Adversarial Training

The Hard Part: Prediction Under Uncertainty

Y LeCun

- Invariant prediction: The training samples are merely representatives of a whole set of possible outputs (e.g. a manifold of outputs).

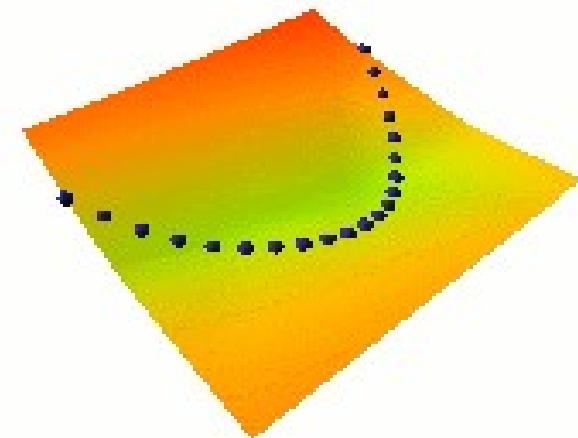
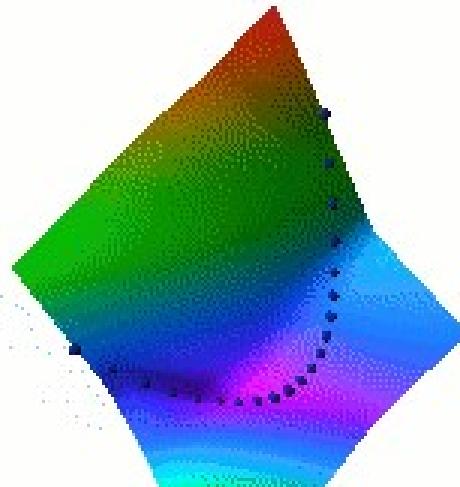
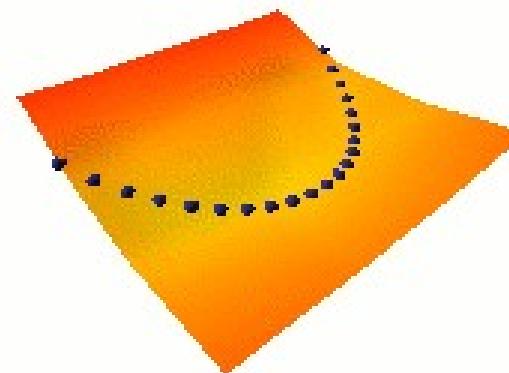
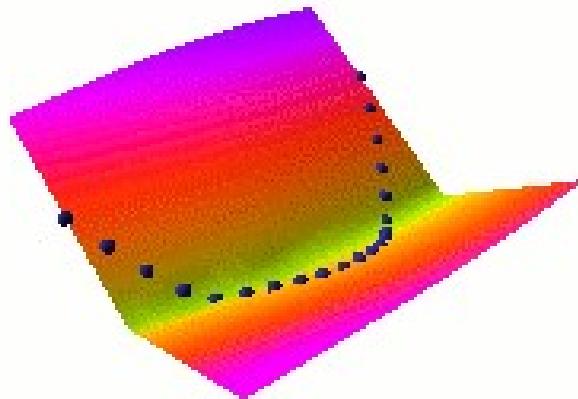




Energy-Based Unsupervised Learning

Y LeCun

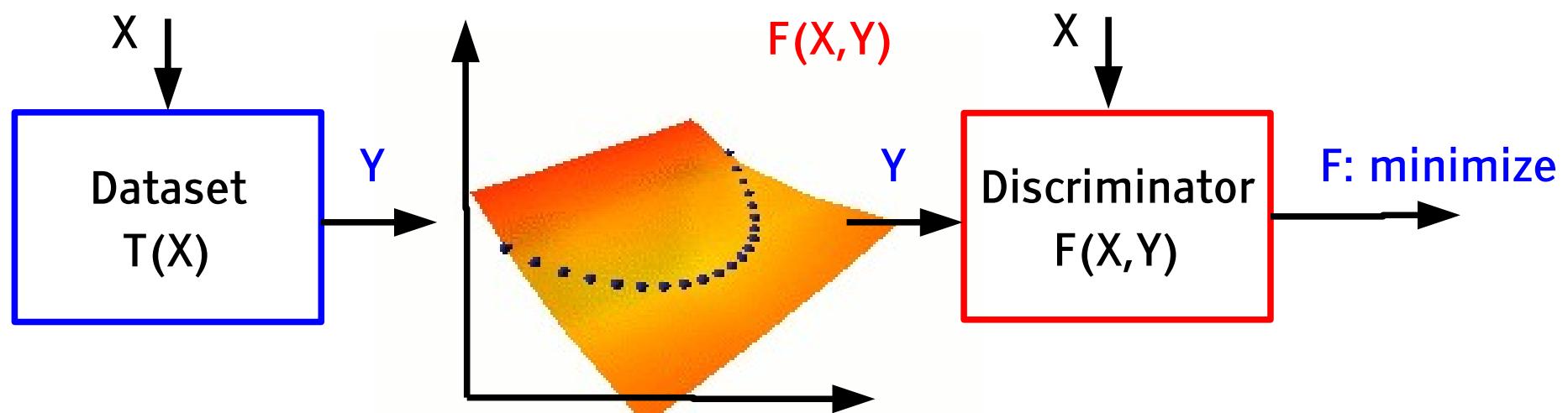
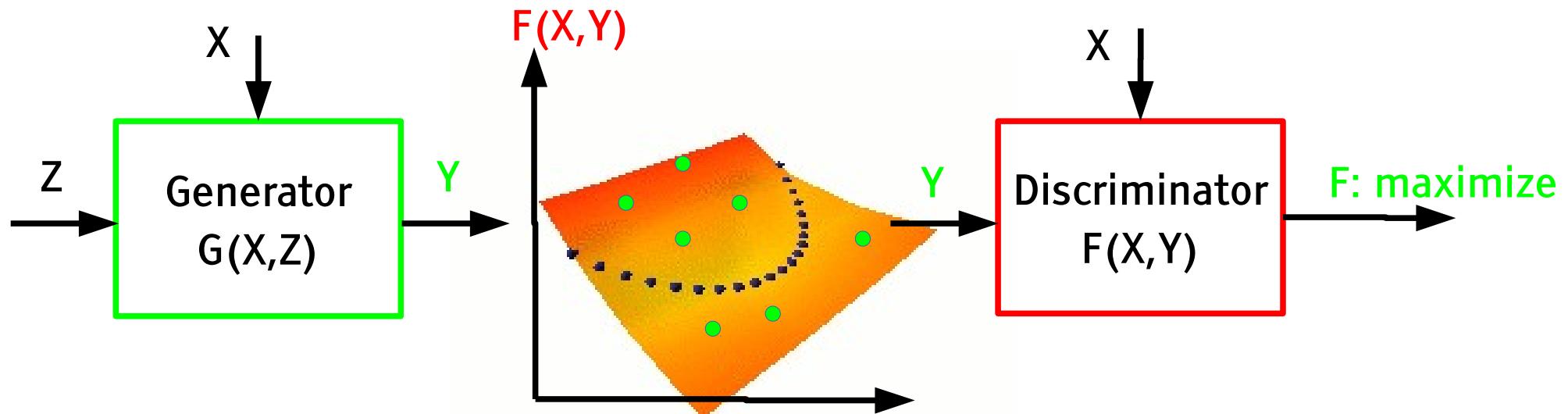
- Energy Function: Takes low value on data manifold, higher values everywhere else
- Push down on the energy of desired outputs. Push up on everything else.
- But how do we choose where to push up?





Adversarial Training: A Trainable Objective Function

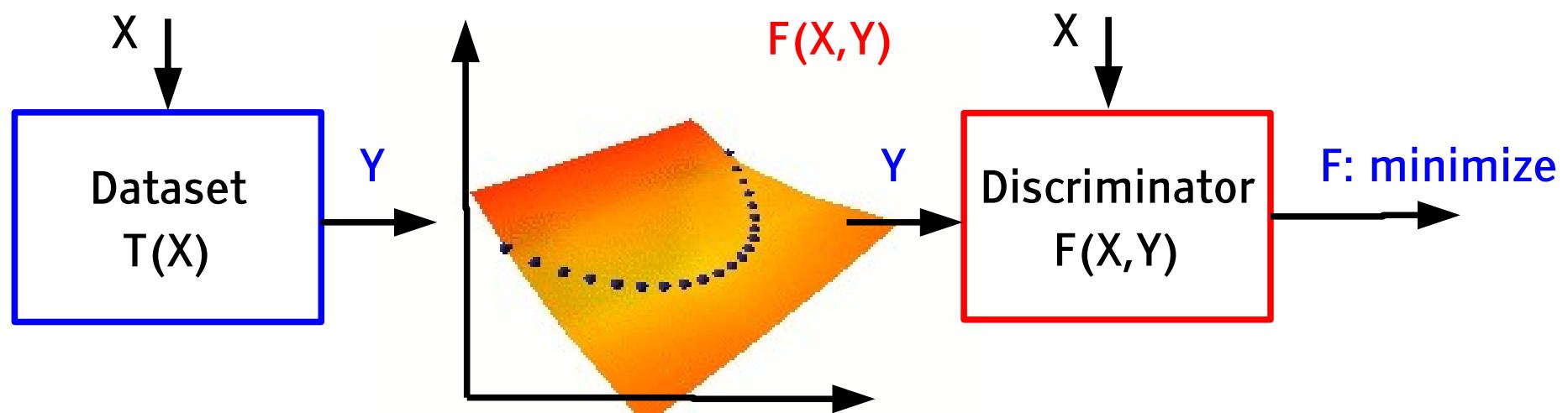
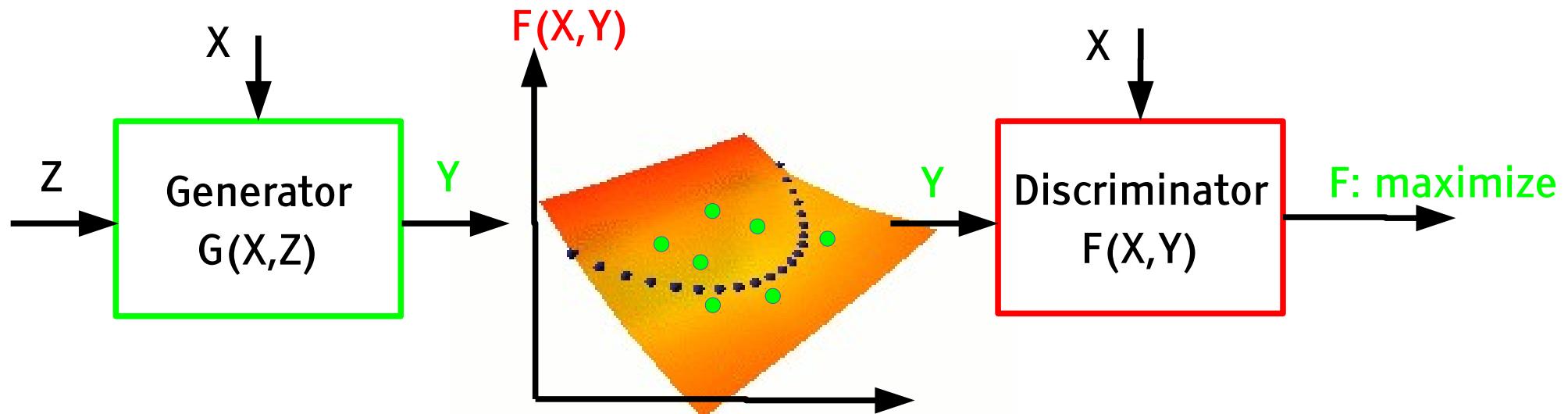
- Adversarial Training [Goodfellow et al. NIPS 2014]
- Energy-based view of adversarial training: generator picks points to push up





Adversarial Training: A Trainable Objective Function

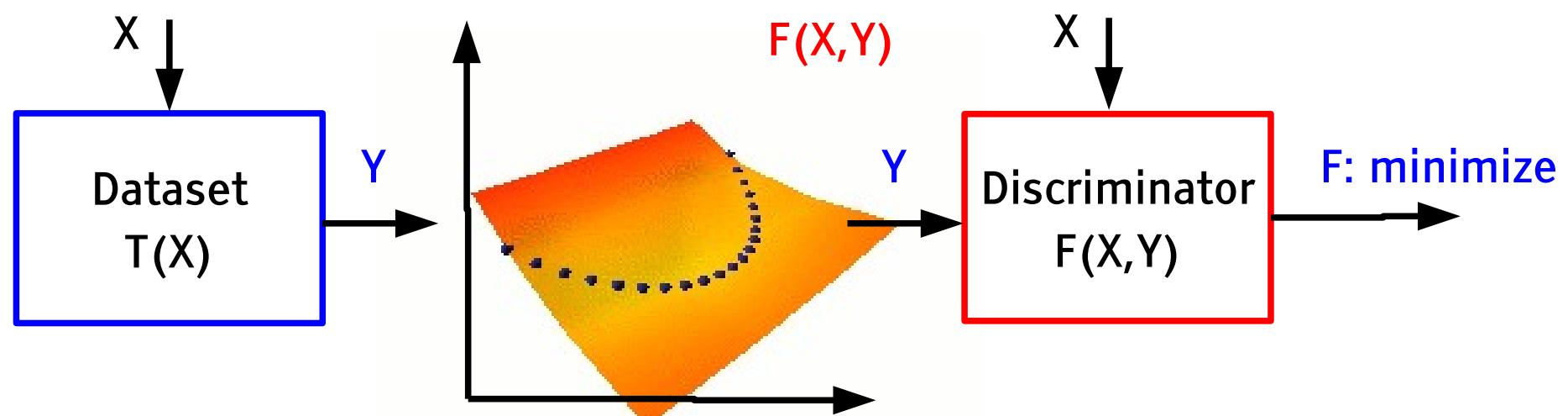
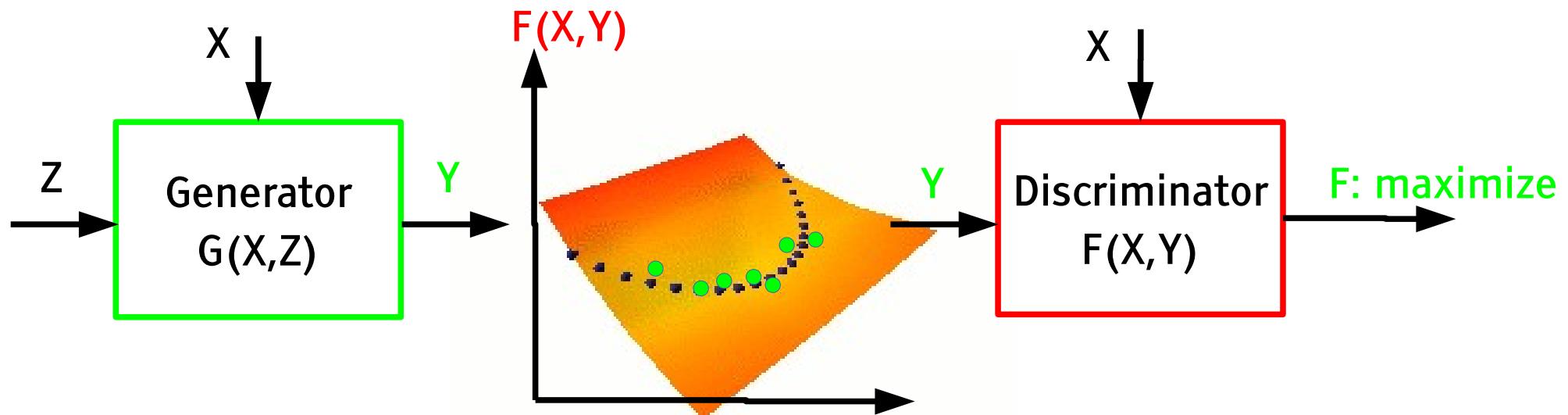
- Adversarial Training [Goodfellow et al. NIPS 2014]
- Energy-based view of adversarial training: generator picks points to push up





Adversarial Training: A Trainable Objective Function

- Energy-based GAN [Zhao, Mathieu, LeCun: arXiv:1609.03.126]

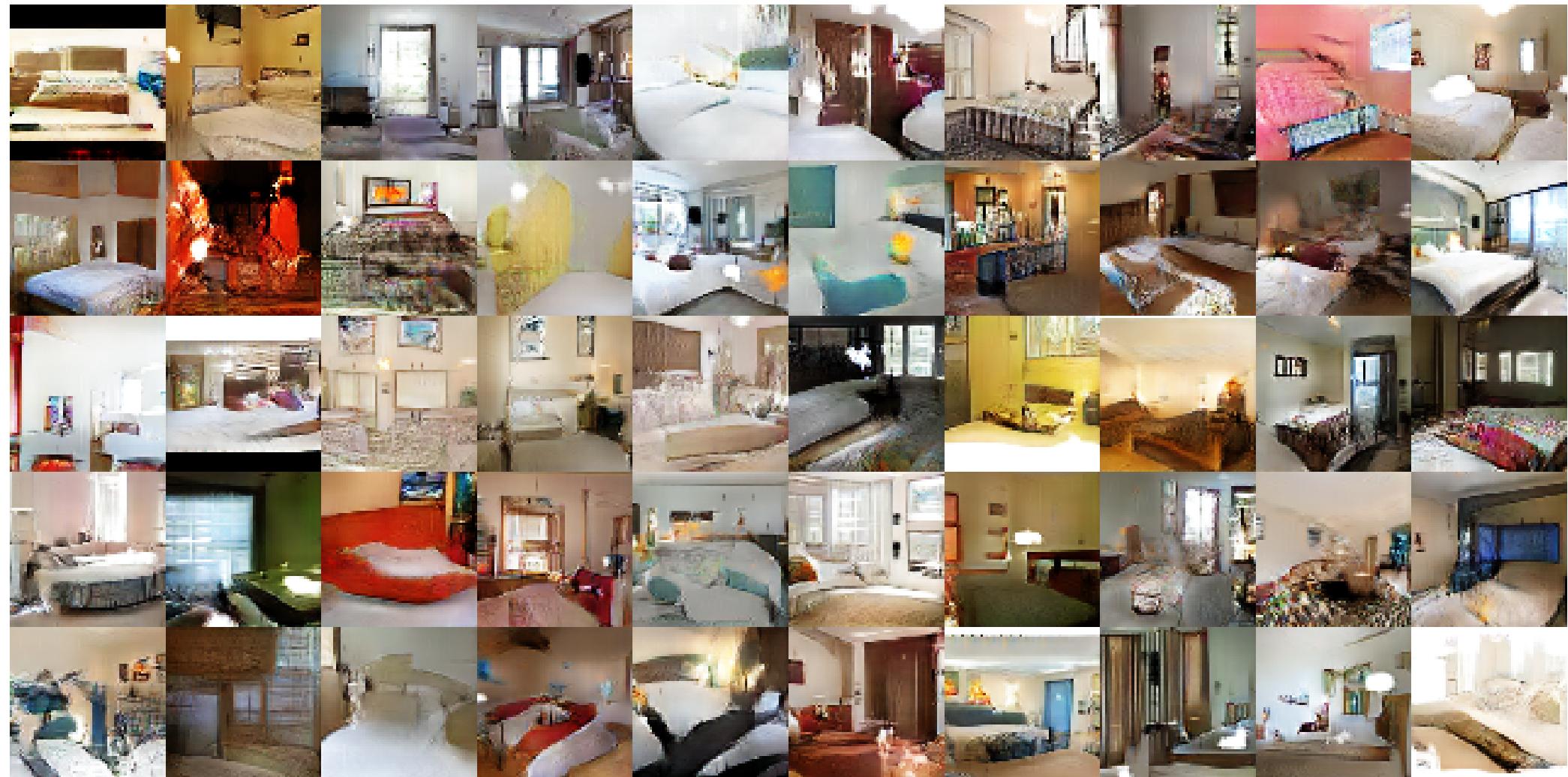




DCGAN: “reverse” ConvNet maps random vectors to images

Y LeCun

- DCGAN: adversarial training to generate images.
- [Radford, Metz, Chintala 2015]
 - ▶ Input: random numbers; output: bedrooms.



Navigating the Manifold



■ DCGAN:
adversarial training
to generate
images.



■ Trained on Manga
characters



■ Interpolates
between character:





Face Algebra (in DCGAN space)

Y LeCun

- DCGAN: adversarial training to generate images.
 - ▶ [Radford, Metz, Chintala 2015]



man
with glasses



man
without glasses



woman
without glasses

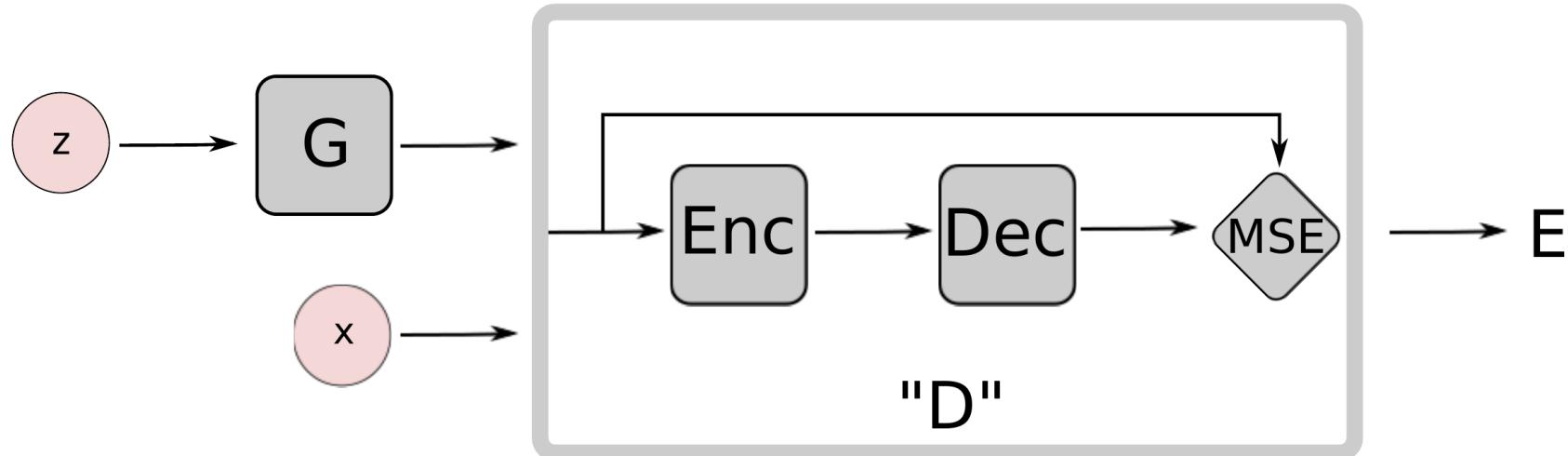


woman with glasses



Energy-Based GAN [Zhao, Mathieu, LeCun: arXiv:1609.03.126]

- Architecture: discriminator is an auto-encoder



- Loss functions

$$f_D(x, z) = D(x) + [m - D(G(z))]^+$$

$$= \|Dec(Enc(x)) - x\| + [m - \|Dec(Enc(G(z))) - G(z)\|]^+,$$

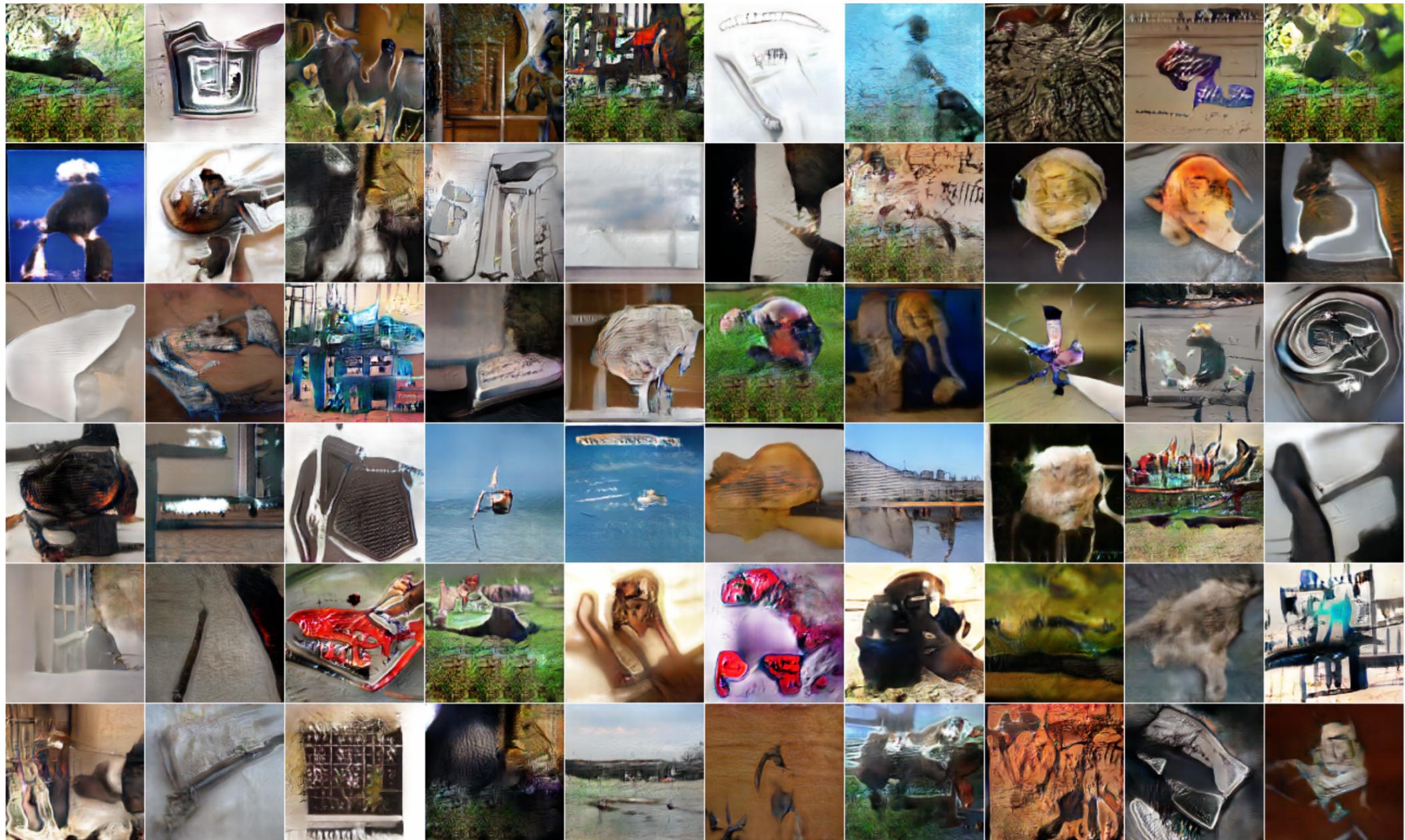
$$f_G(z) = \|D(G(z))\|$$

$$= \|Dec(Enc(G(z))) - G(z)\|$$



Energy-Based GAN trained on ImageNet at 128x128 pixels

Y LeCun

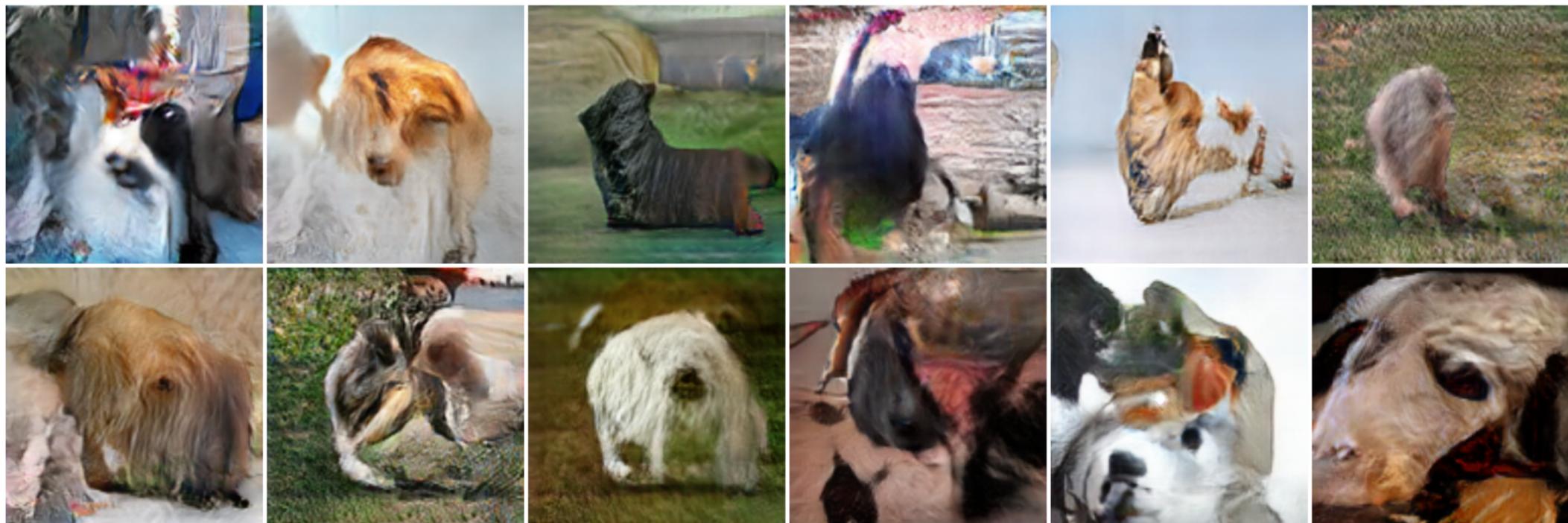




Energy-Based GAN trained on ImageNet at 256x256 pixels

Y LeCun

■ Trained on dogs



Video Prediction (with adversarial training)

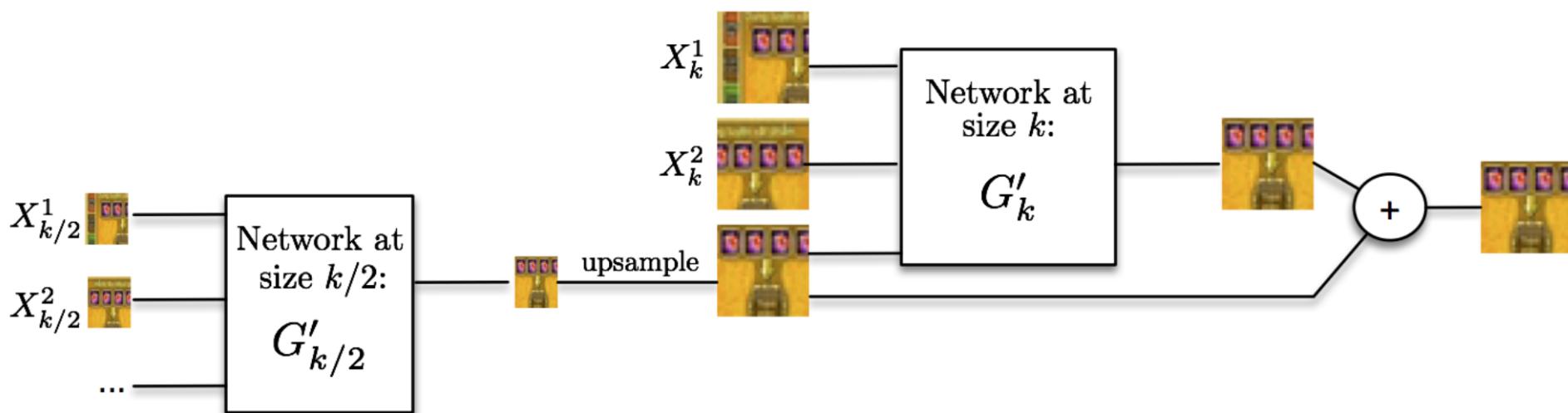
[Mathieu, Couprie, LeCun ICLR 2016]
arXiv:1511:05440

Multi-Scale ConvNet for Video Prediction

- 4 to 8 frames input → ConvNet → 1 to 8 frames output
- Multi-scale ConvNet, without pooling
- If trained with least square: blurry output



Predictor (multiscale ConvNet Encoder-Decoder)



Predictive Unsupervised Learning

Y LeCun

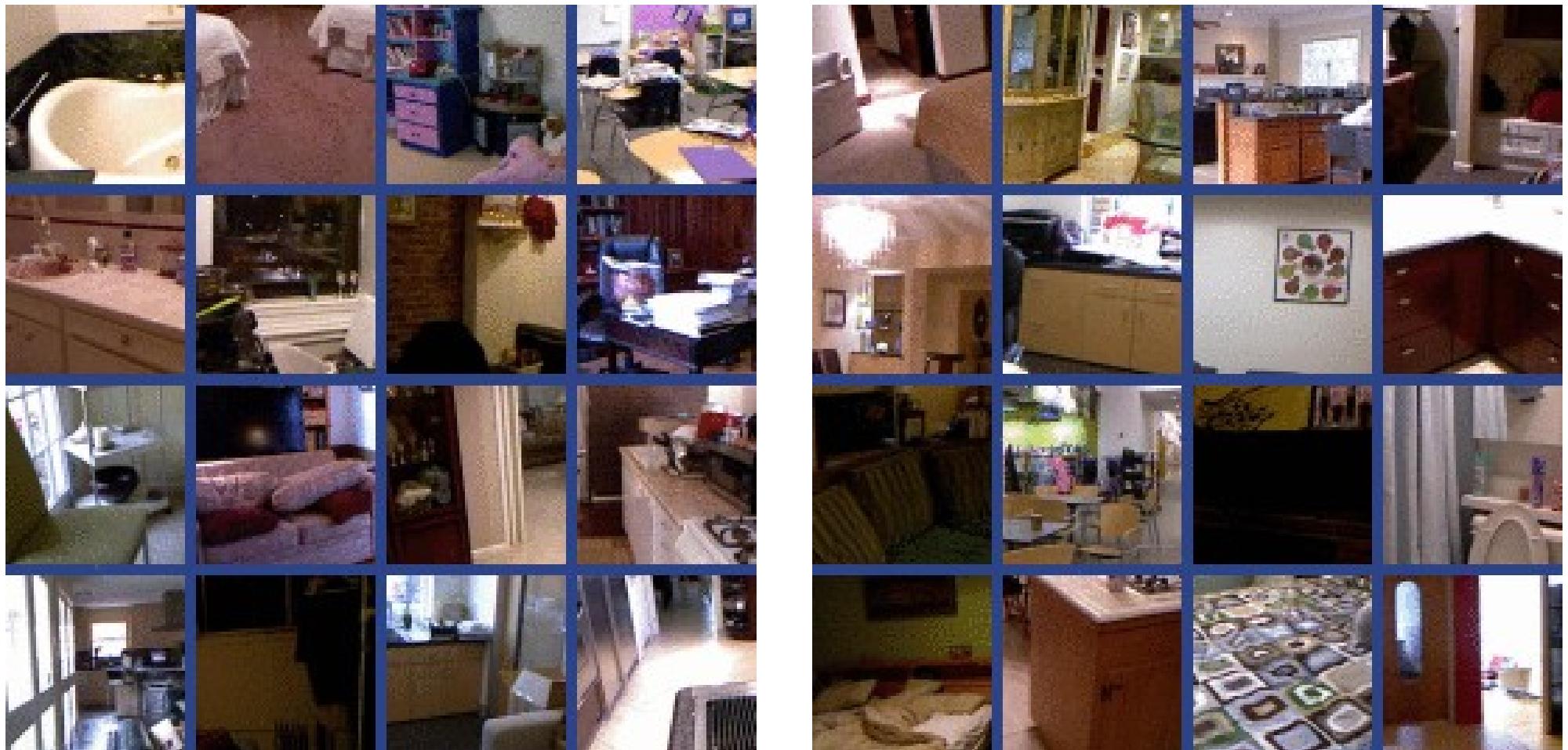
- Our brains are “prediction machines”
- Can we train machines to predict the future?
- Some success with “adversarial training”
 - ▶ [Mathieu, Couprie, LeCun arXiv:1511:05440]
- But we are far from a complete solution.





Video Prediction: predicting 5 frames

Y LeCun





Video Prediction: predicting 5 frames

Y LeCun





Video Prediction: predicting 50 frames

Y LeCun



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).

What will future AI systems be like?

Y LeCun

■ Human and animal behavior has basic “drives” hardwired by evolution

- ▶ Fight/flight, hunger, self-preservation, pain avoidance, desire for social interaction, etc...

■ Humans do bad things to each other because of these drives (mostly)

- ▶ Violence under threat, desire for material resource and social power...

■ But an AI system will not have these drives unless we build them into it.

■ It's difficult for us humans to imagine an intelligent entity without these drives

- ▶ But they are specific to humans
- ▶ We have plenty of different forms of intelligence in the animal world

