Large-Scale Deep Learning

IPAM SUMMER SCHOOL

Marc'Aurelio Ranzato - Google

ranzato@google.com

www.cs.toronto.edu/~ranzato

UCLA, 24 July 2012

Two Approches to Deep Learning

Deep Neural Nets:

- usually more efficient (at training & test time)
- typically more unconstrained (partition function has to be replaced by other constraints, e.g. sparsity).
- more flexible
- ideal for end-to-end learning of complex systems

Deep Probabilistic Models:

- typically intractable
- easier to compose
- easier to interpret (e.g., you can generate samples from them)
- better deal with uncertainty

Example: Auto-Encoder

Neural Net:

code $Z = \sigma(W_e^T X + b_e)$

reconstruction $\hat{X} = W_d Z + b_d$

Probabilistic Model (Gaussian RBM):

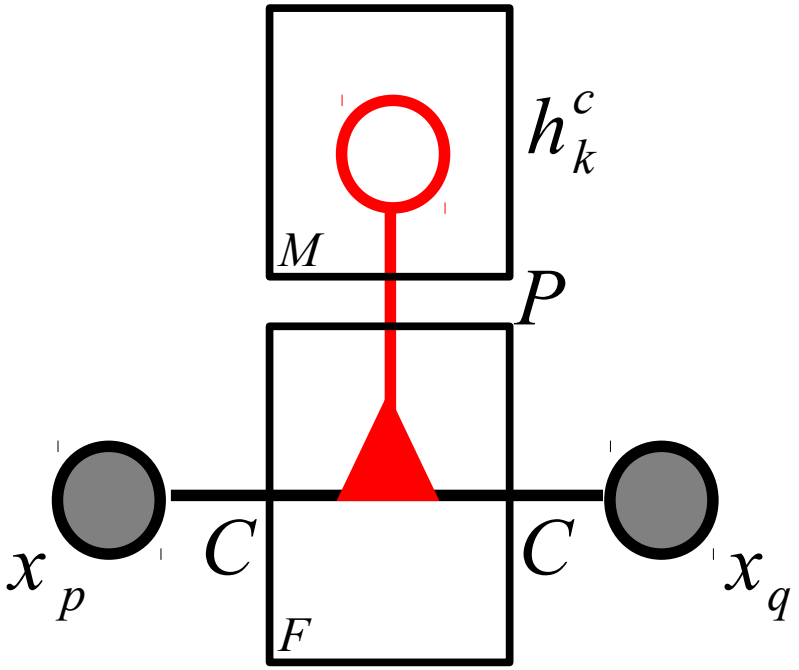
$$E[Z|X] = \sigma(W^T X + b_e)$$

$$E[X|Z] = W Z + b_d$$

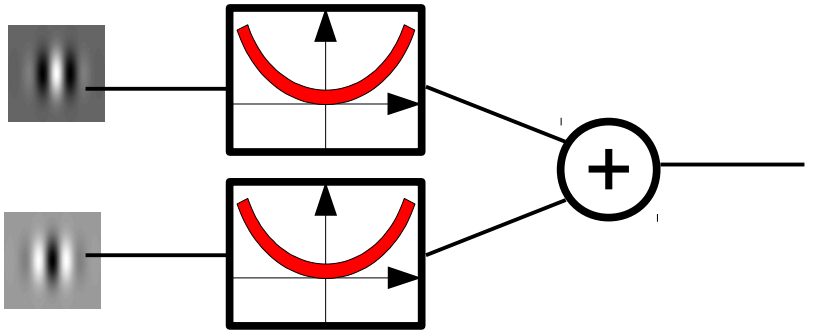
Vincent "A connection between score matching and denoising autoencoders" Neural Comp. 2011

Swersky et al. "On autoencoders and score matching for EBM" ICML 2011

gated MRF



$$p(h_k^c = 1 | v) = f(P_k (C'v)^2)$$



- relation to simple-complex cell model

gated MRF & Subspace ICA

Subspace ICA:

$$p(h_k^c = 1 | v) = f\left(P_k (C'v)^2\right)$$

Hyvarinen et al. "Emergence of…features by...independent feature subspaces" Neural Comp 2000

Probabilistic Model (gated MRF):

$$p(h_k^c = 1 | v) = \sigma\left(-P_k (C'v)^2\right)$$

Welling et al. "Learning sparse topographic representations with PoT" NIPS 2002

Ranzato et al. "Factored 3-way RBMs for modeling natural images" AISTATS 2010

gated MRF & IPSD

IPSD:

$$feature = \sqrt{(P_k (C'v)^2)}$$

Kavukcuoglu et al. "Learning invariant features through topographic filter maps" CVPR 2009

Probabilistic Model (gated MRF):

$$p(h_k^c = 1 | v) = \sigma(-P_k (C'v)^2)$$

Welling et al. "Learning sparse topographic representations with PoT" NIPS 2002

Ranzato et al. "Factored 3-way RBMs for modeling natural images" AISTATS 2010

Probabilities: yes/no?

Deep Neural Nets:

- usually more efficient (at training & test time)
- typically more unconstrained (partition function has to be replaced by other constraints, e.g. sparsity).
- more flexible
- ideal for end-to-end learning of complex systems

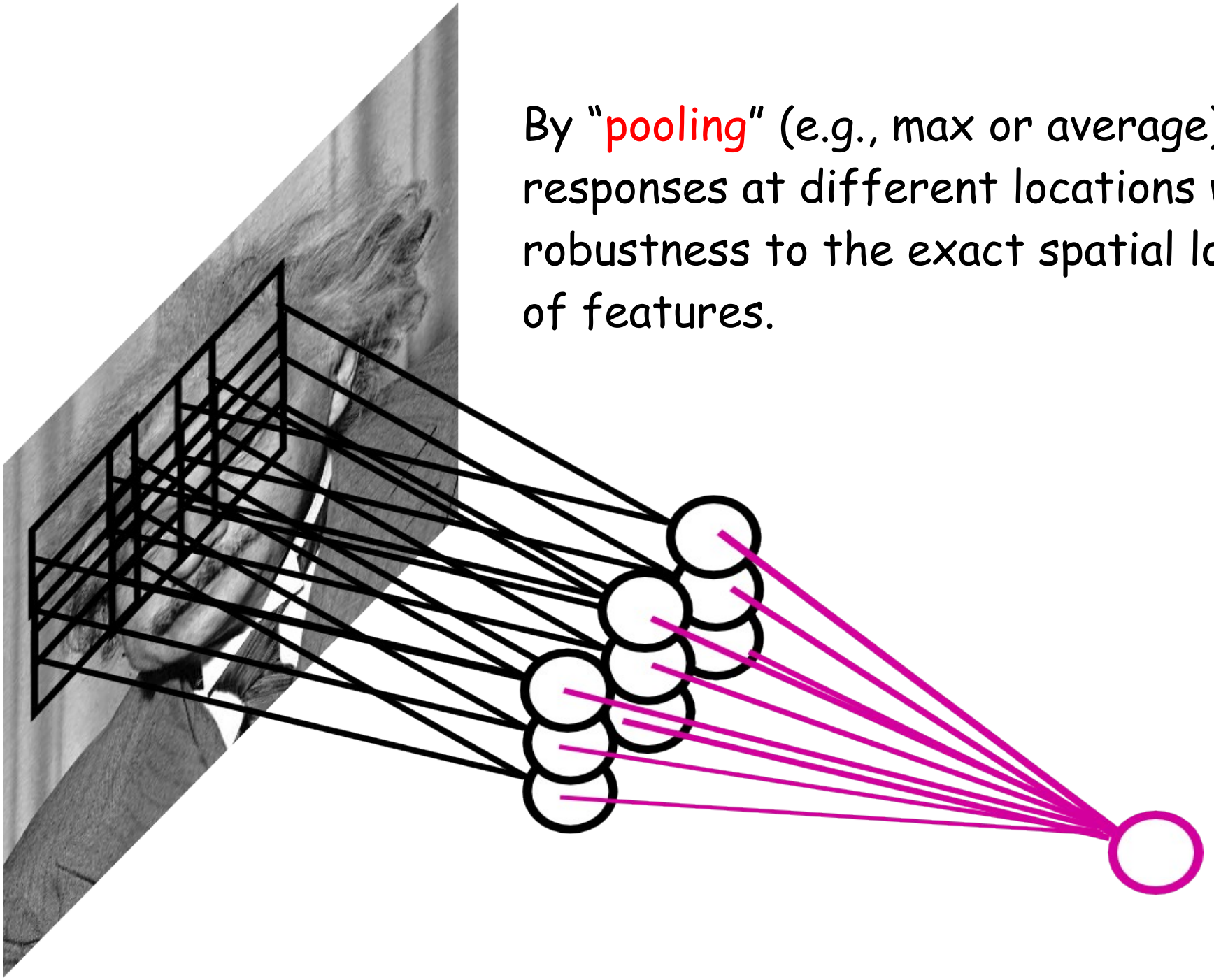
Deep Probabilistic Models:

- typically intractable
- easier to compose
- easier to interpret (e.g., you can generate samples from them)
- better deal with uncertainty

Today we are going to focus on
Deep Neural Nets
since they are more easily scalable

POOLING

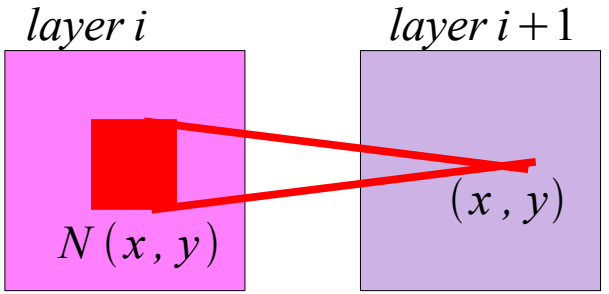
By "pooling" (e.g., max or average) filter responses at different locations we gain robustness to the exact spatial location of features.



POOLING

Over the years, some new modules have proven to be very effective when plugged into conv-nets:

- L2 Pooling



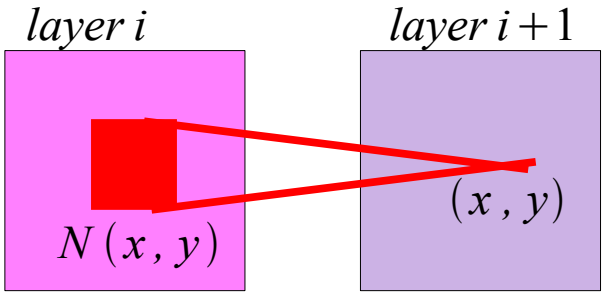
$$h_{i+1,x,y} = \sqrt{\sum_{(j,k)\in N(x,y)} h_{i,j,k}^2}$$

Jarrett et al. "What is the best multi-stage architecture for object recognition?" ICCV 2009

POOLING

Over the years, some new modules have proven to be very effective when plugged into conv-nets:

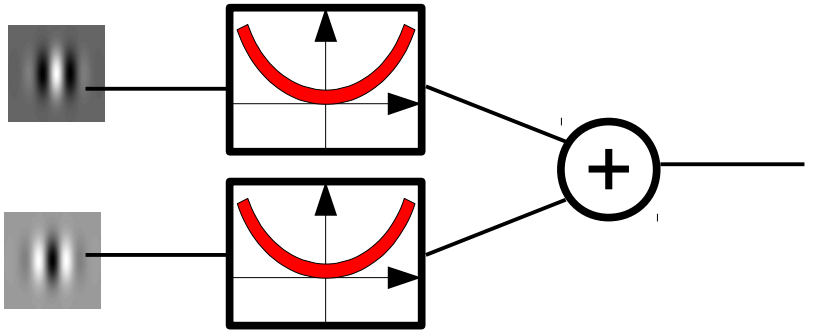
- L2 Pooling



$$h_{i+1,x,y} = \sqrt{\sum_{(j,k)\in N(x,y)} h^2_{i,j,k}}$$

Like a gated MRF!

$$p(h^c_k = 1|v) = f(P_k(C'v)^2)$$

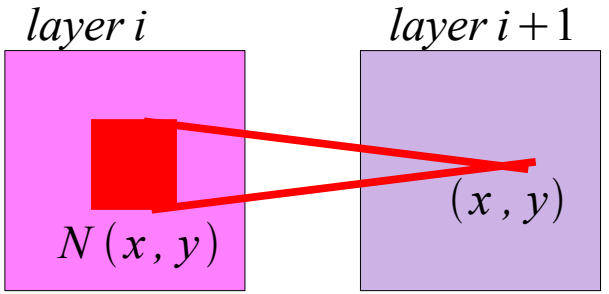


Jarrett et al. "What is the best multi-stage architecture for object recognition?" ICCV 2009

POOLING & LCN

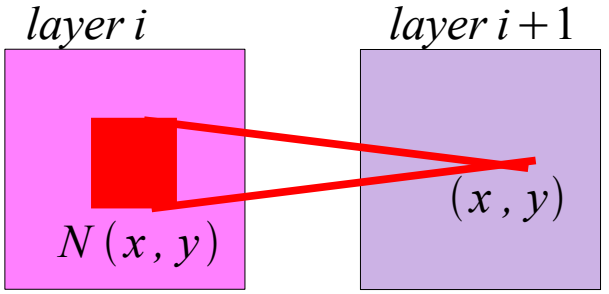
Over the years, some new modules have proven to be very effective when plugged into conv-nets:

- L2 Pooling



$$h_{i+1,x,y} = \sqrt{\sum_{(j,k)\in N(x,y)} h_{i,j,k}^2}$$

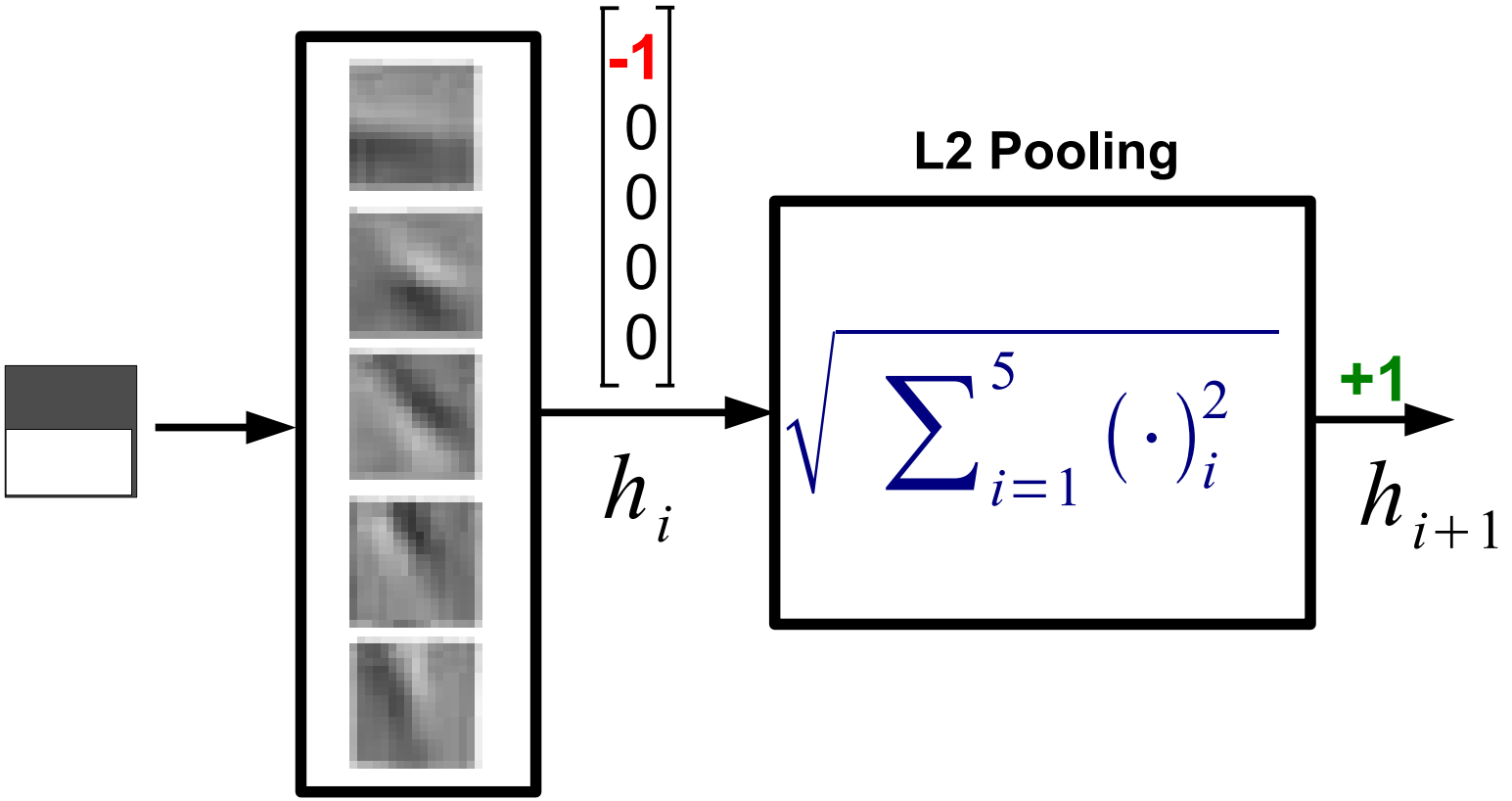
- Local Contrast Normalization



$$h_{i+1,x,y} = \frac{h_{i,x,y} - m_{i,N(x,y)}}{\sigma_{i,N(x,y)}}$$

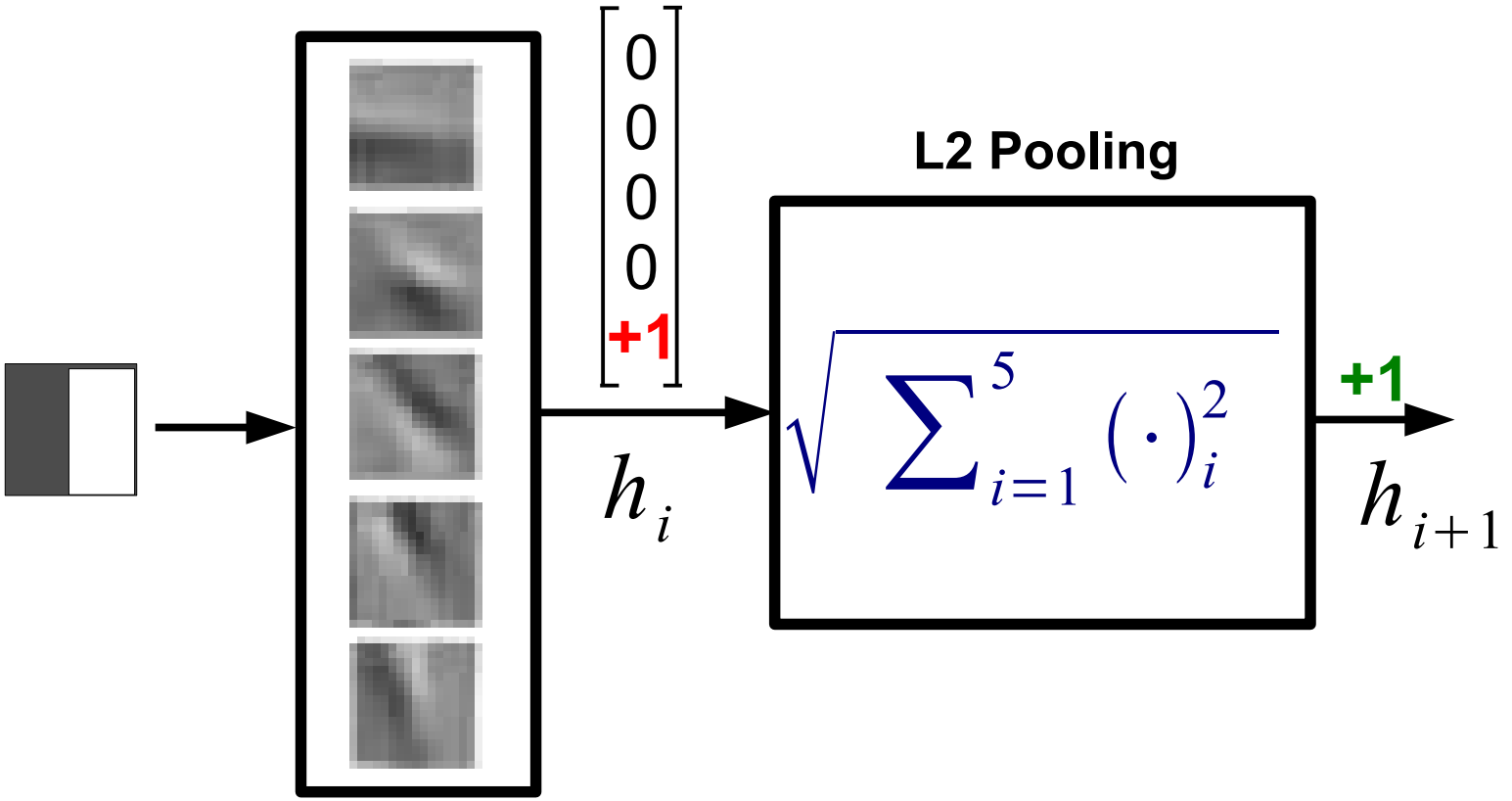
Jarrett et al. "What is the best multi-stage architecture for object recognition?" ICCV 2009

L2 POOLING



Kavukguoglu et al. "Learning invariant features ..." CVPR 2009

L2 POOLING

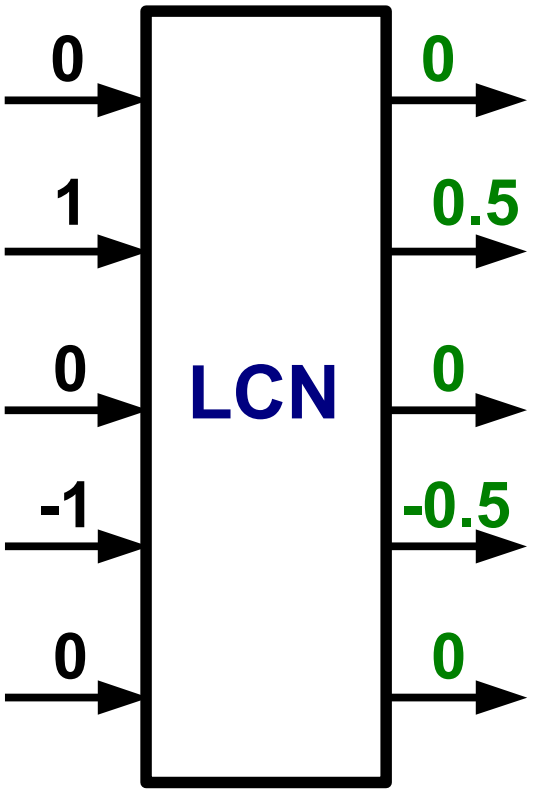


Kavukguoglu et al. "Learning invariant features ..." CVPR 2009

L2 Pooling helps learning
representations more robust to local distortions!

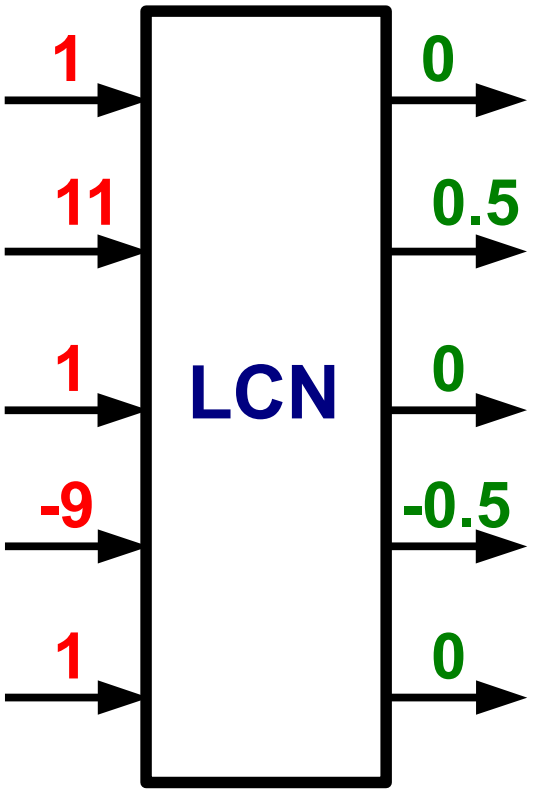
LOCAL CONTRAST NORMALIZATION

$$h_{i+1,x,y} = \frac{h_{i,x,y} - m_{i,N(x,y)}}{\sigma_{i,N(x,y)}}$$



LOCAL CONTRAST NORMALIZATION

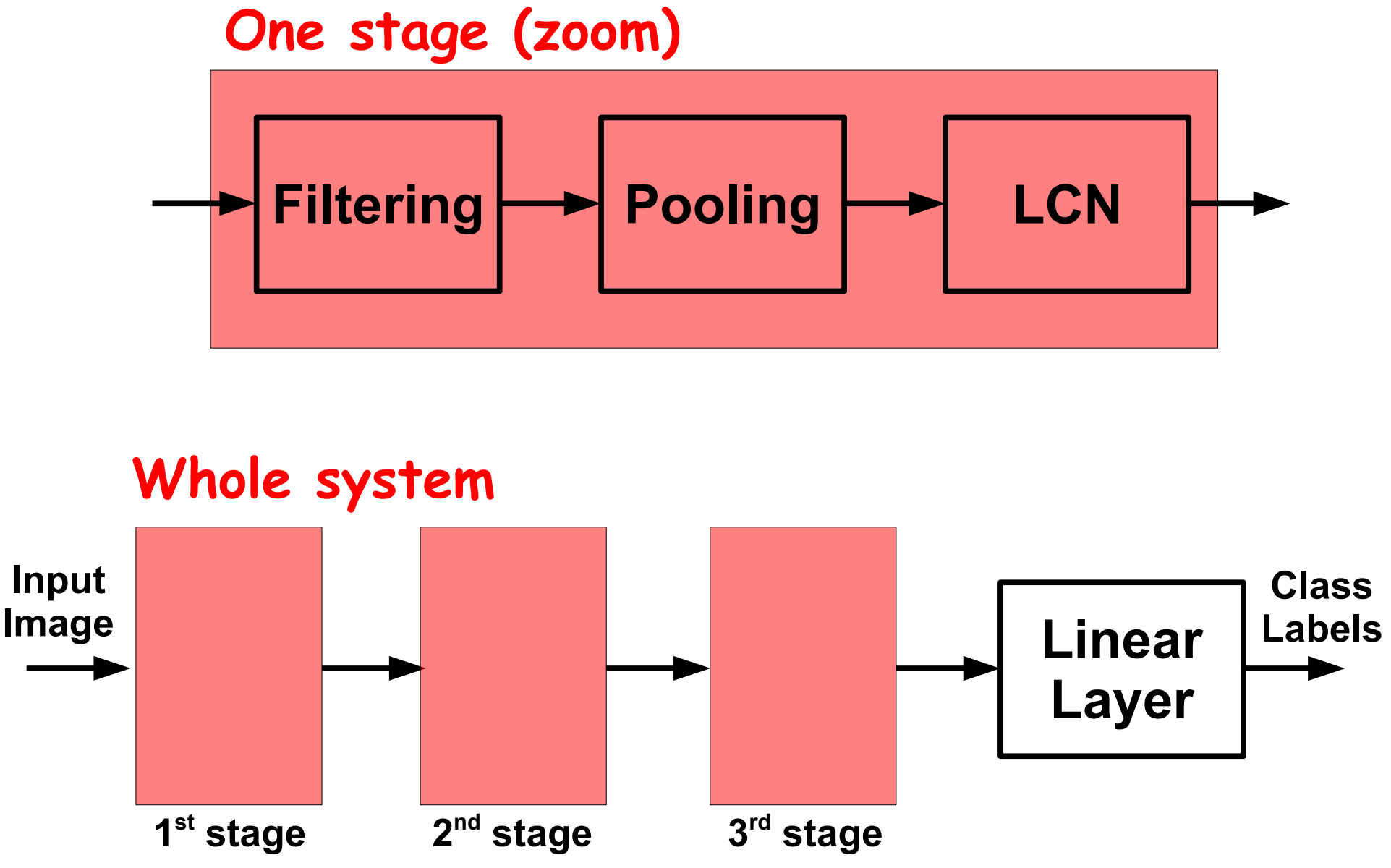
$$h_{i+1,x,y} = \frac{h_{i,x,y} - m_{i,N(x,y)}}{\sigma_{i,N(x,y)}}$$



L2 POOLING & LCN

L2 Pooling & Local Contrast Normalization
help learning more invariant representations!

CONV NETS: TYPICAL ARCHITECTURE



CONV NETS: TRAINING

Since convolutions and sub-sampling are differentiable, we can use standard back-propagation.

Algorithm:

Given a small mini-batch

- FPROP

- BPROP

- PARAMETER UPDATE

CONV NETS: EXAMPLES

- **Object category recognition**

  Boureau et al. "Ask the locals: multi-way local pooling for image recognition" ICCV 2011

- **Segmentation**

  Turaga et al. "Maximin learning of image segmentation" NIPS 2009

- **OCR**

  Ciresan et al. "MCDNN for Image Classification" CVPR 2012

- **Pedestrian detection**

  Kavukcuoglu et al. "Learning convolutional feature hierarchies for visual recognition" NIPS 2010

- **Robotics**

  Sermanet et al. "Mapping and planning..with long range perception" IROS 2008

LIMITATIONS & SOLUTIONS

- requires lots of labeled data to train
+ unsupervised learning

- difficult optimization
+ layer-wise training

- scalability
+ distributed training

LIMITATIONS & SOLUTIONS

- requires lots of labeled data to train
+ unsupervised learning

- difficult optimization
+ layer-wise training

- scalability
+ distributed training

Tera-Scale Deep Learning @ Google

Observation #1: more features always improve performance unless data is scarce.

Observation #2: deep learning methods have higher capacity and have the potential to model data better.

Q #1: Given lots of data and lots of machines, can we scale up deep learning methods?

Q #2: Will deep learning methods perform much better?

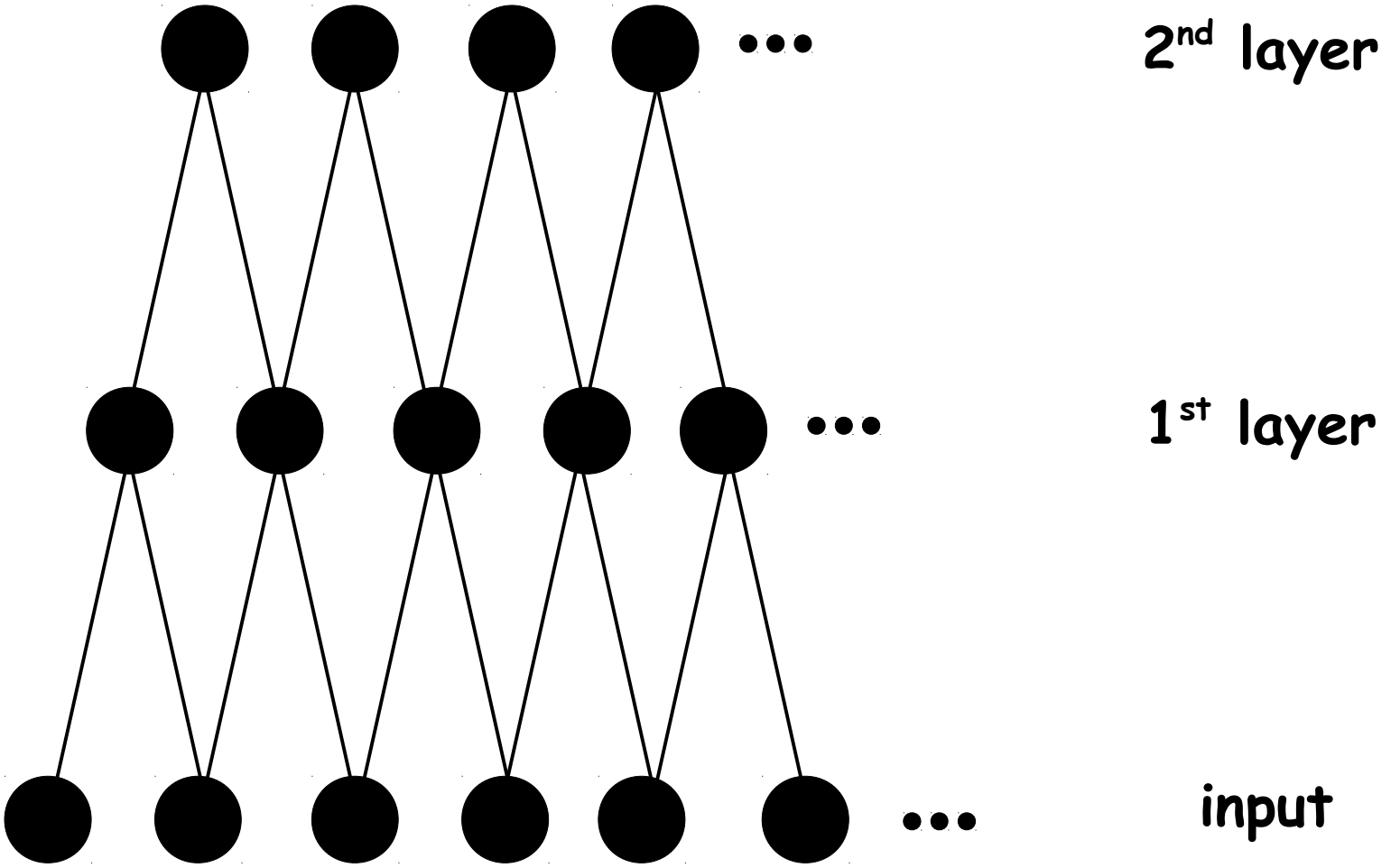
The Challenge

A Large Scale problem has:

- lots of training samples (>10M)
- lots of classes (>10K) and
- lots of input dimensions (>10K).

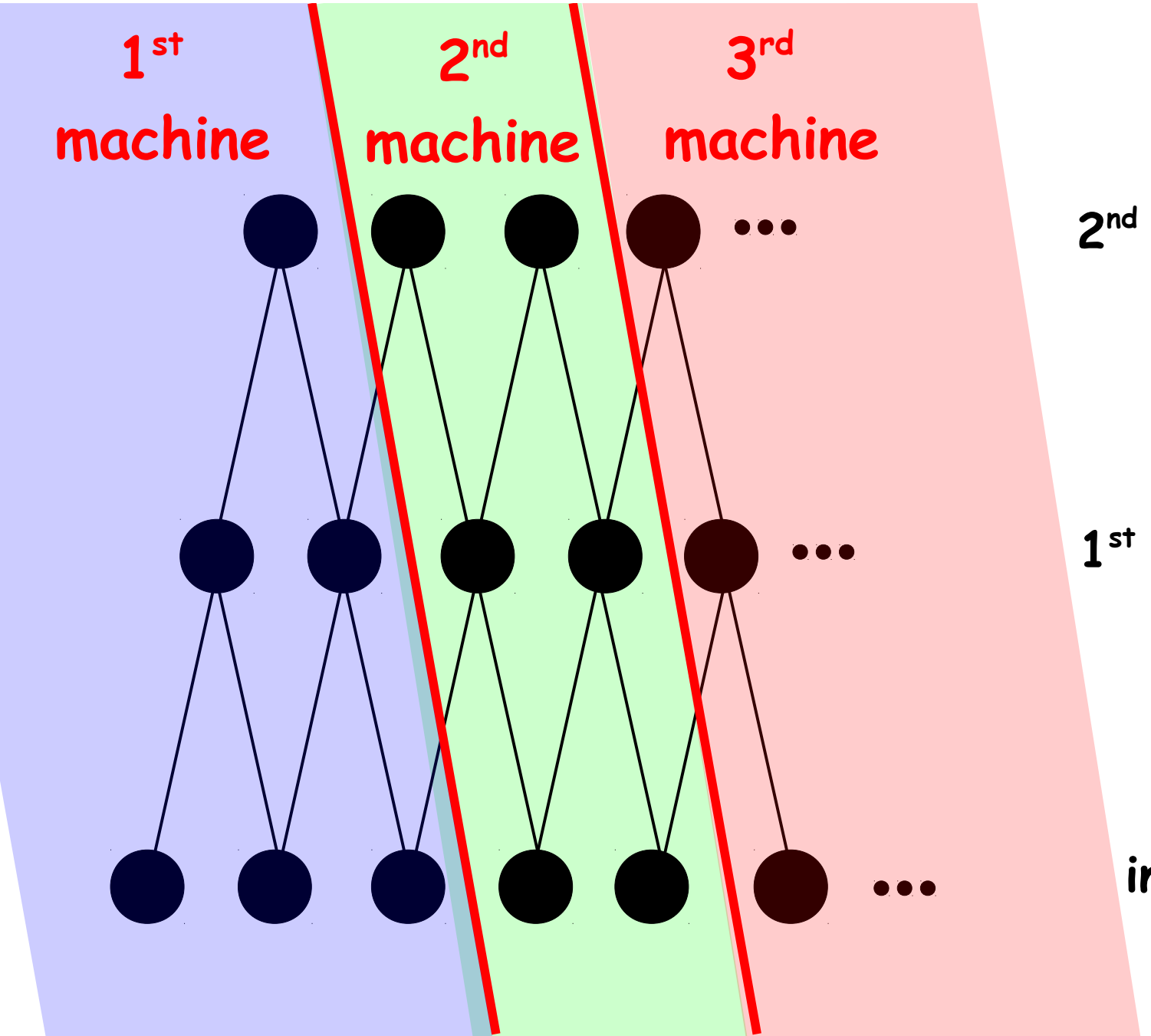
- best optimizer in practice is on-line SGD which is naturally sequential, hard to parallelize.
- layers cannot be trained independently and in parallel, hard to distribute
- model can have lots of parameters that may clog the network, hard to distribute across machines

Our Solution

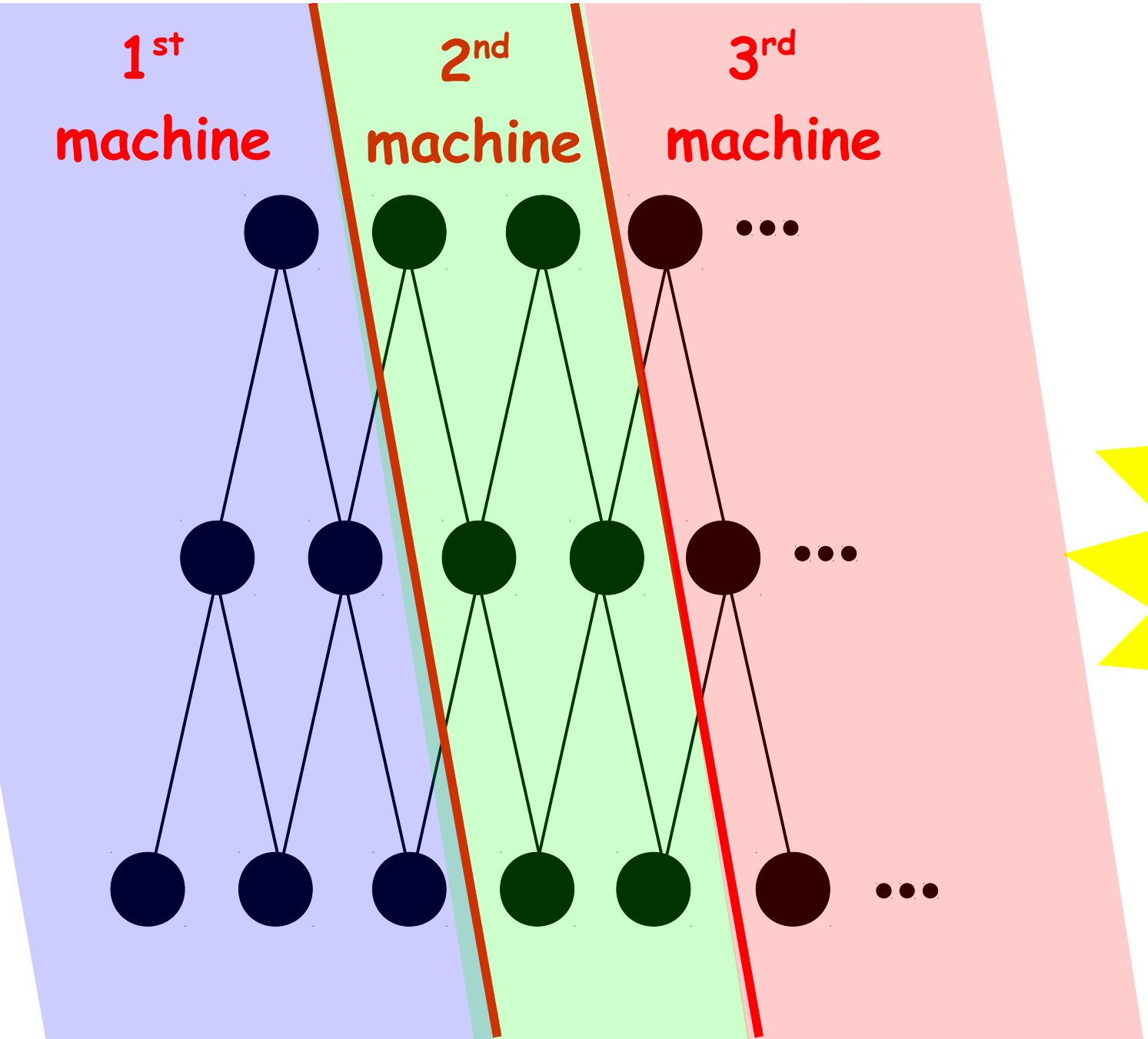


Le et al. "Building high-level features using large-scale unsupervised learning" ICML 2012

Our Solution

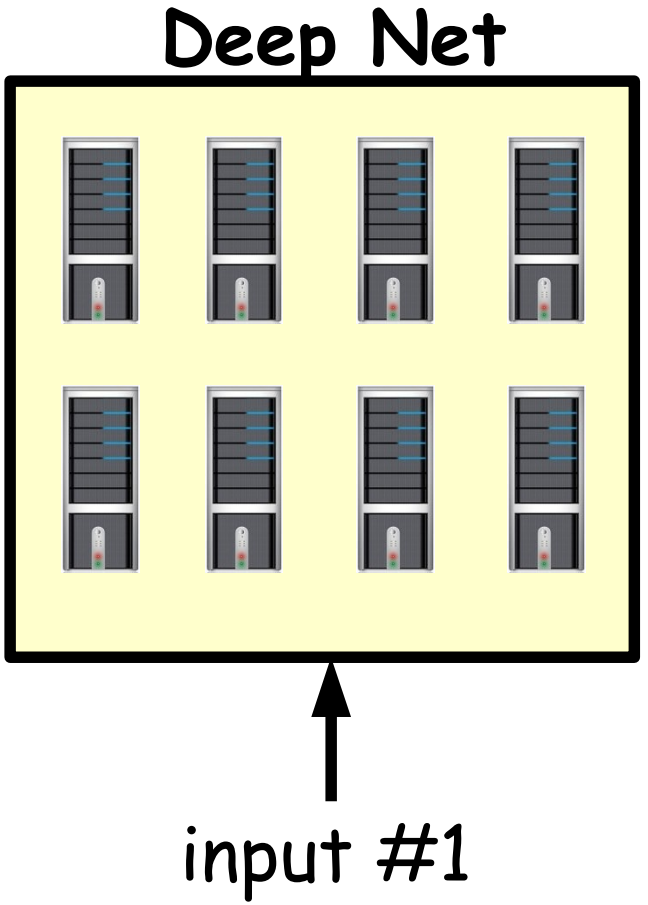


Our Solution



MODEL PARALLELISM

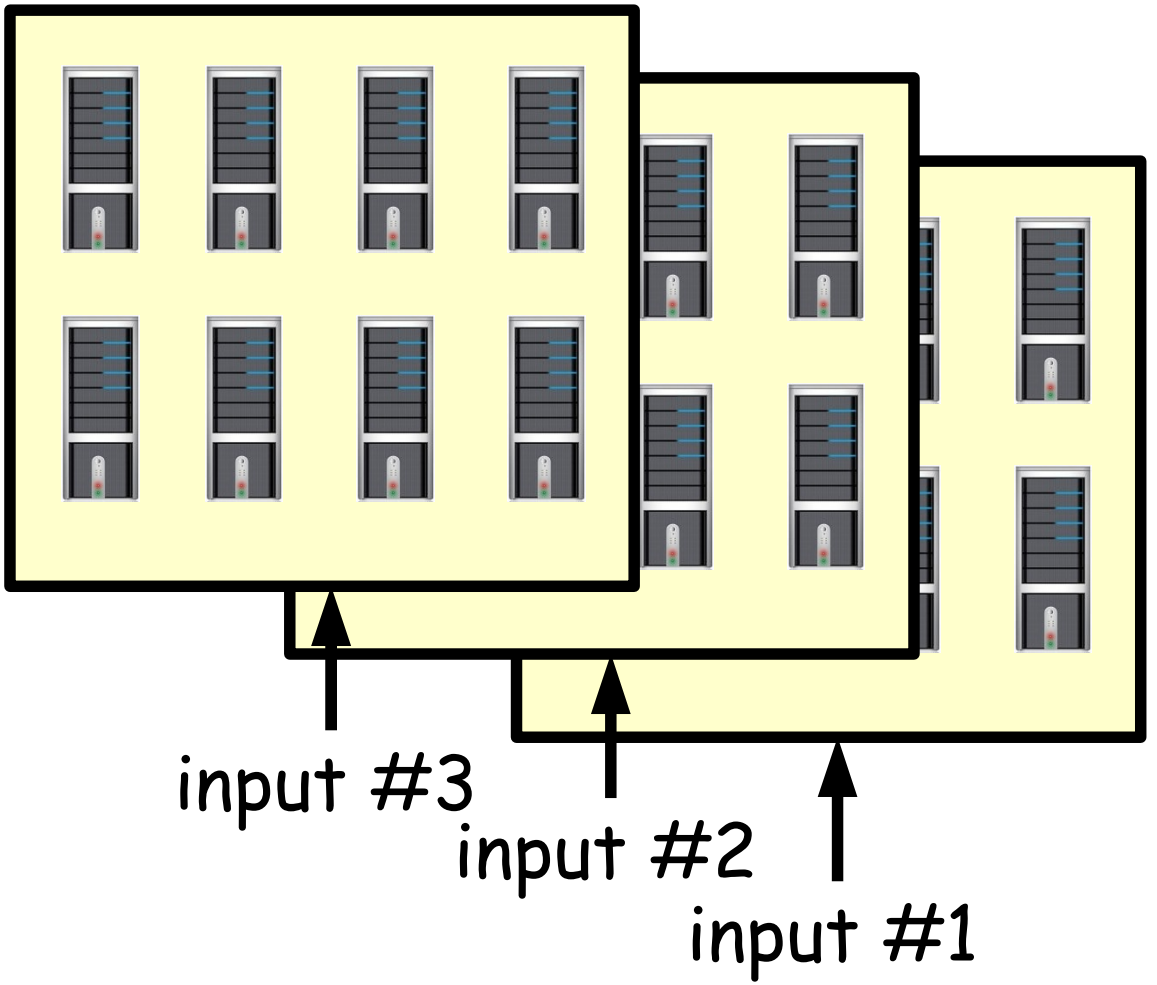
Distributed Deep Nets



MODEL PARALLELISM

Le et al. "Building high-level features using large-scale unsupervised learning" ICML 2012

Distributed Deep Nets



MODEL PARALLELISM

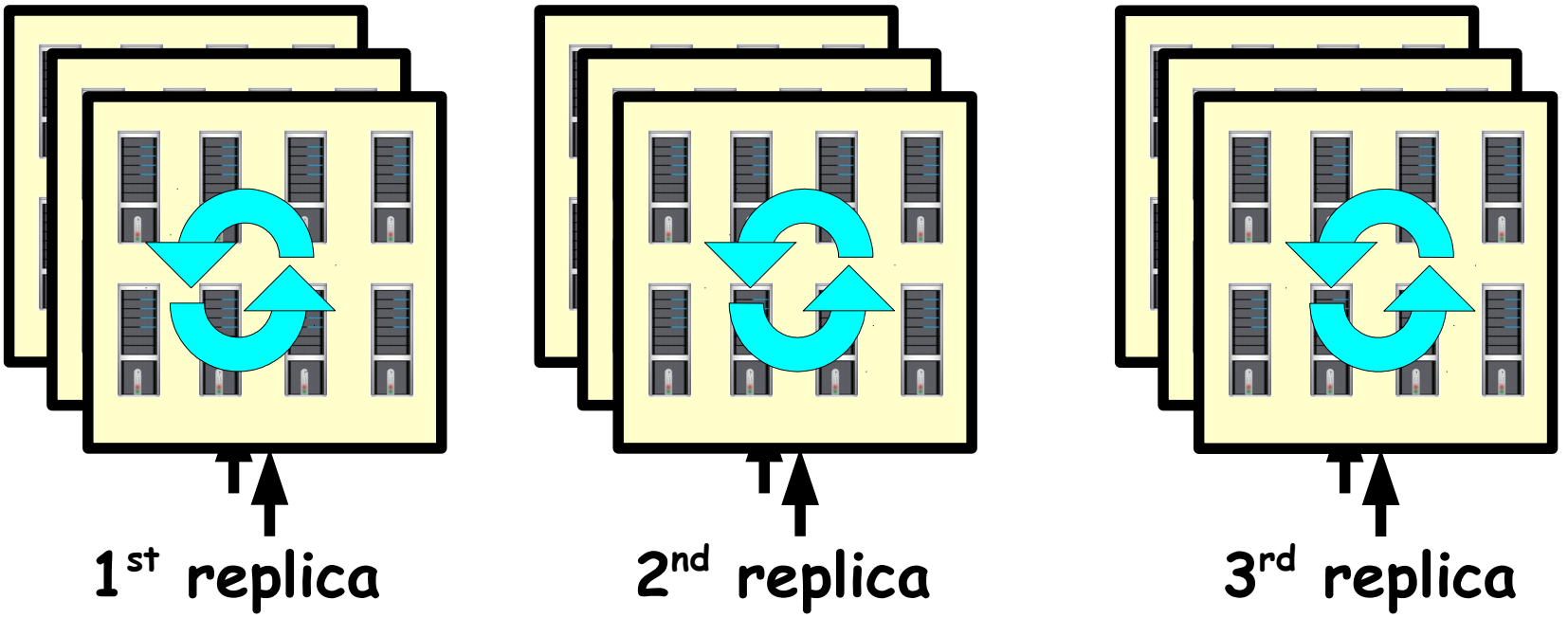
+

DATA PARALLELISM

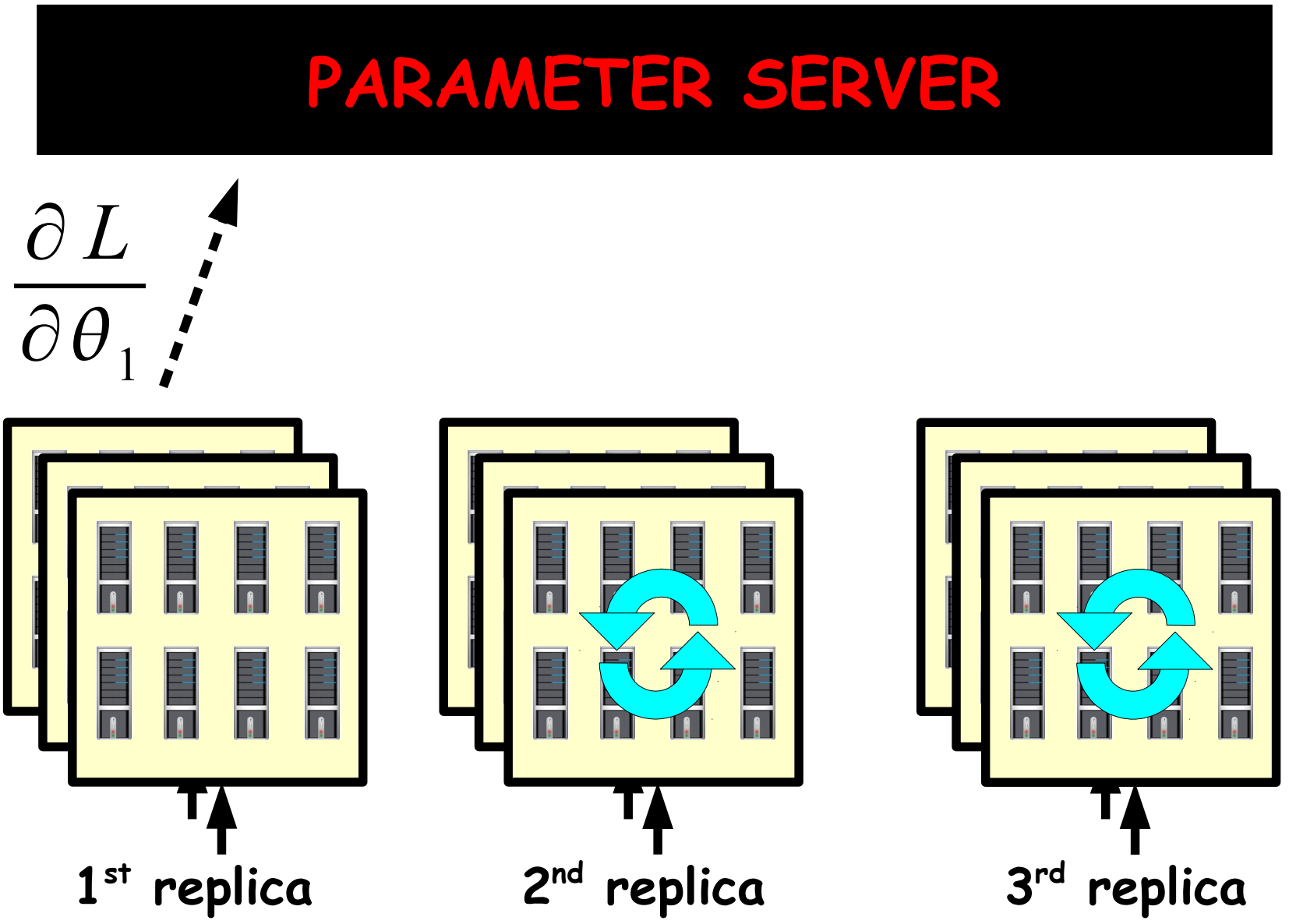
Le et al. "Building high-level features using large-scale unsupervised learning" ICML 2012

Asynchronous SGD

PARAMETER SERVER

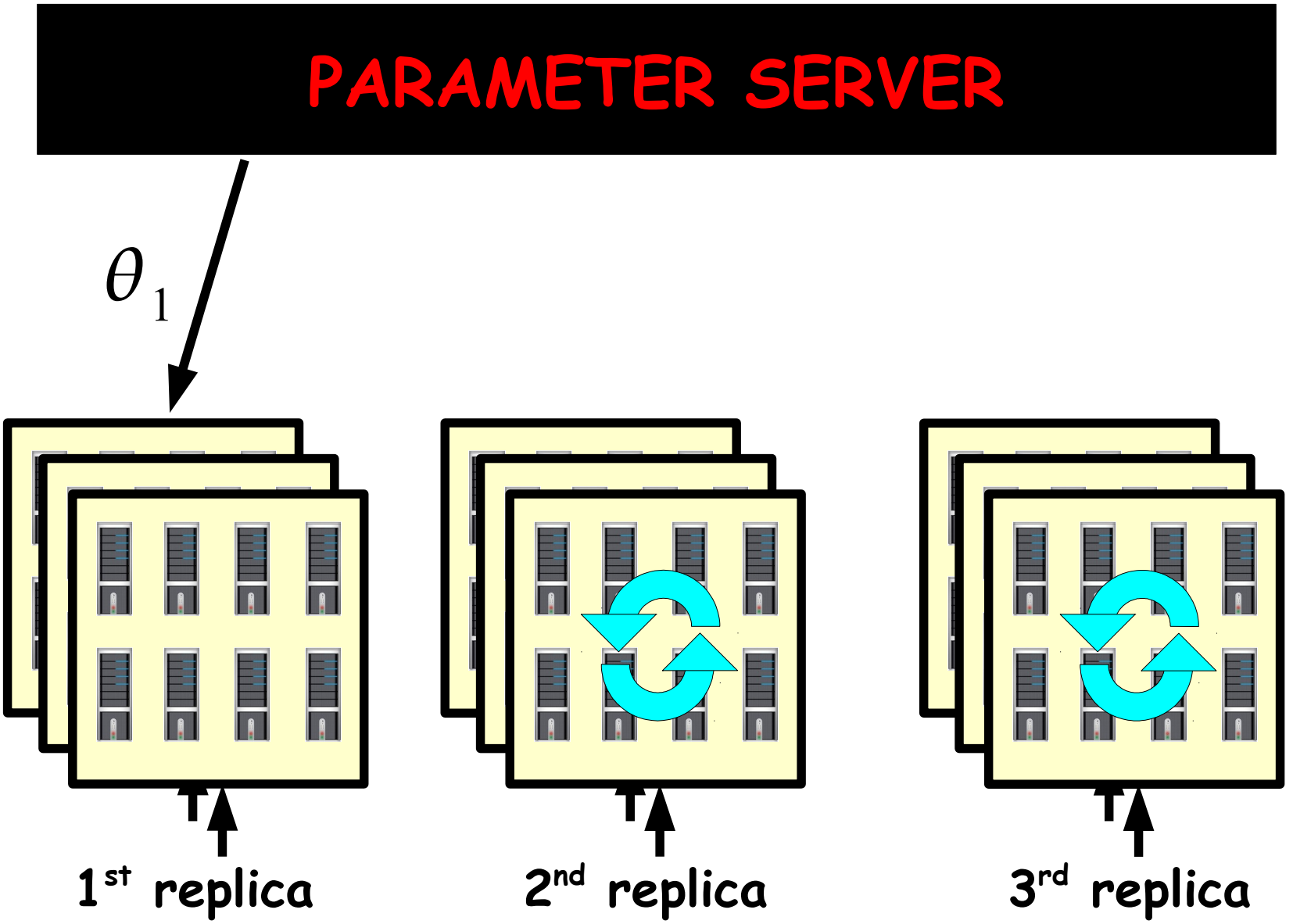


Asynchronous SGD



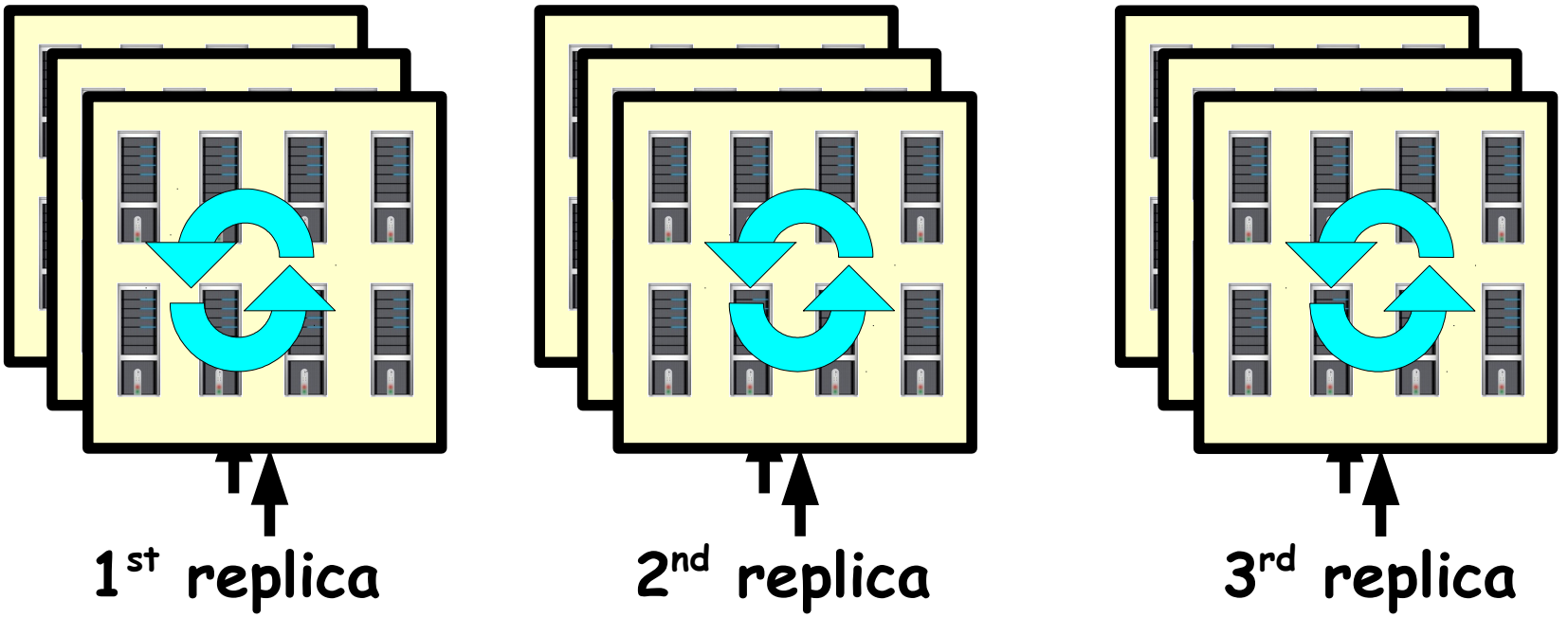
Ranzato

Asynchronous SGD

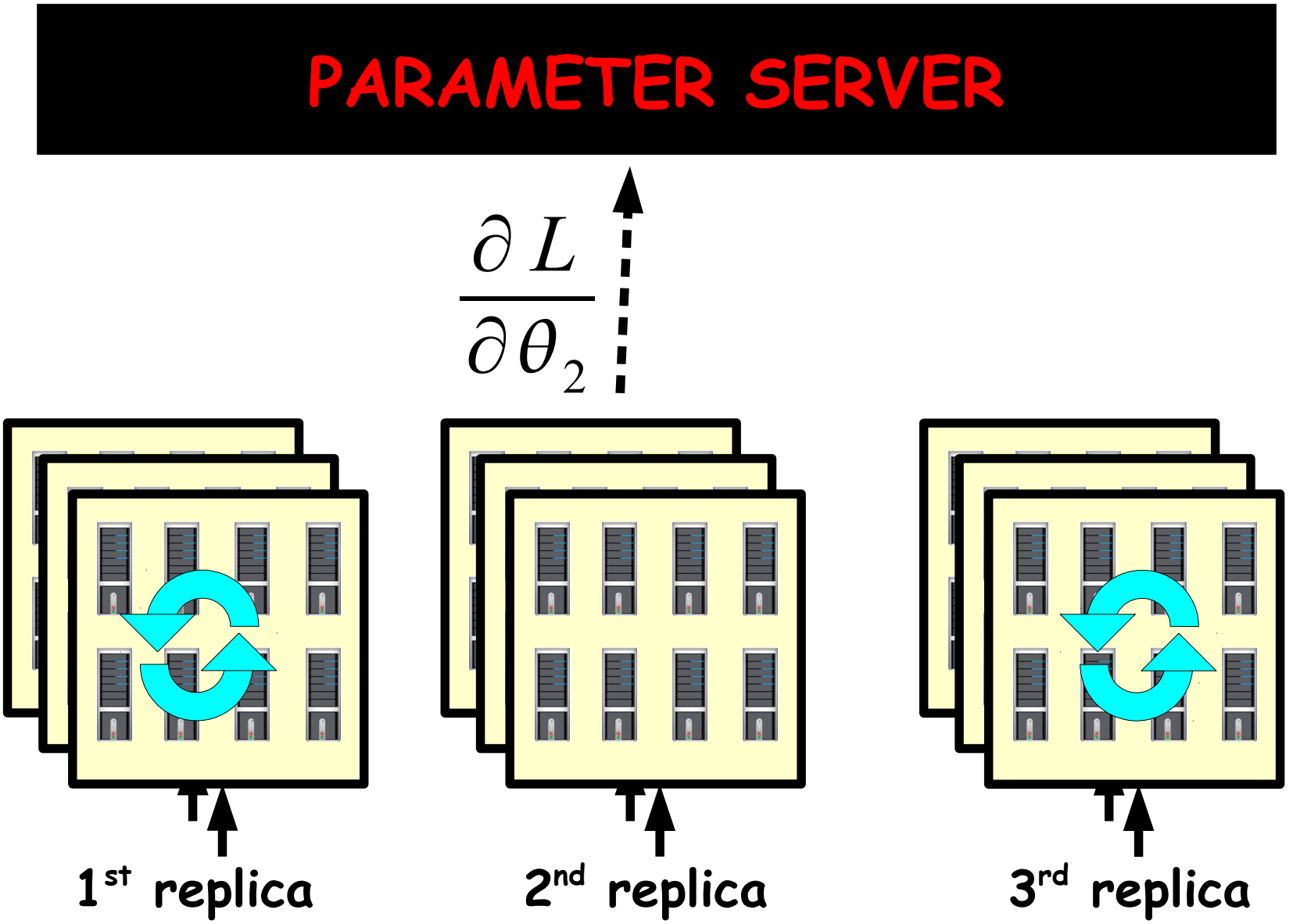


Asynchronous SGD

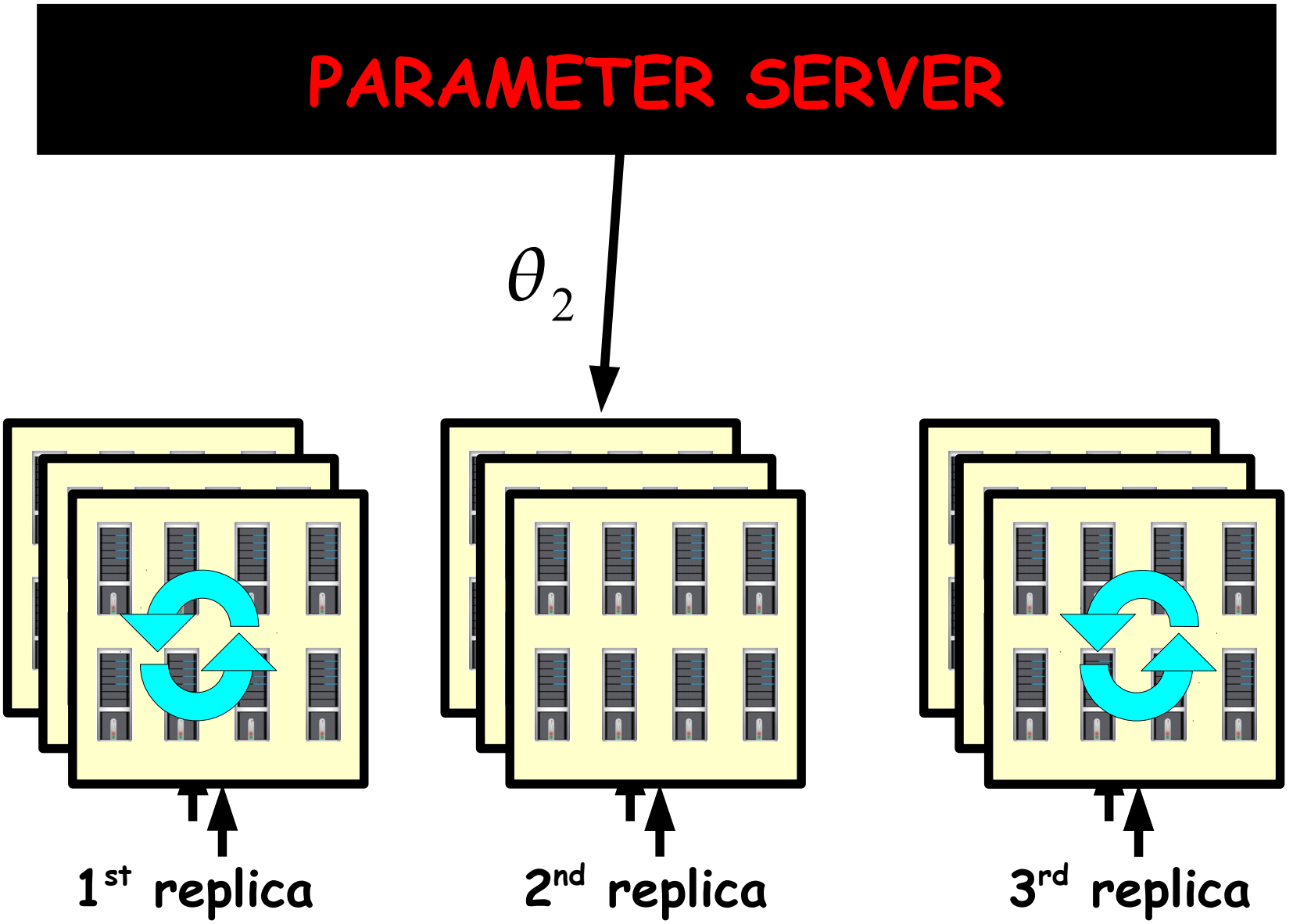
PARAMETER SERVER
(update parameters)



Asynchronous SGD

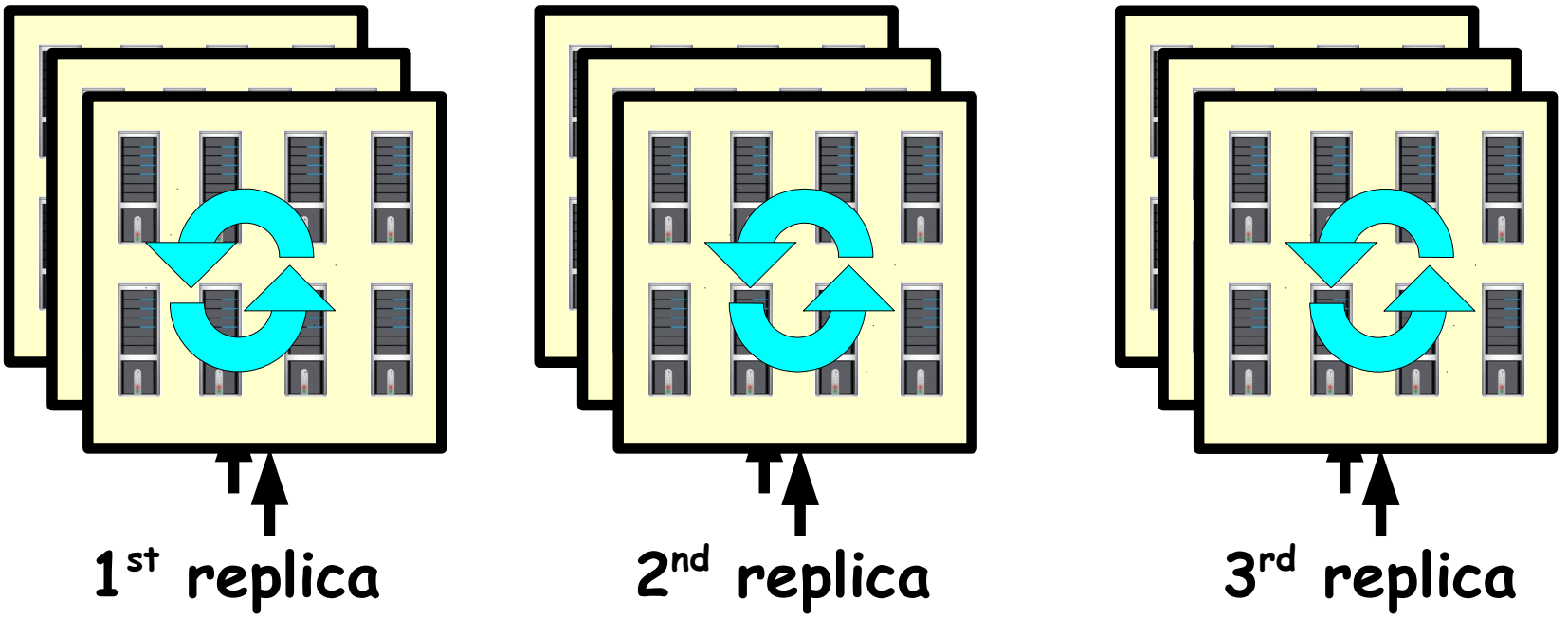


Asynchronous SGD

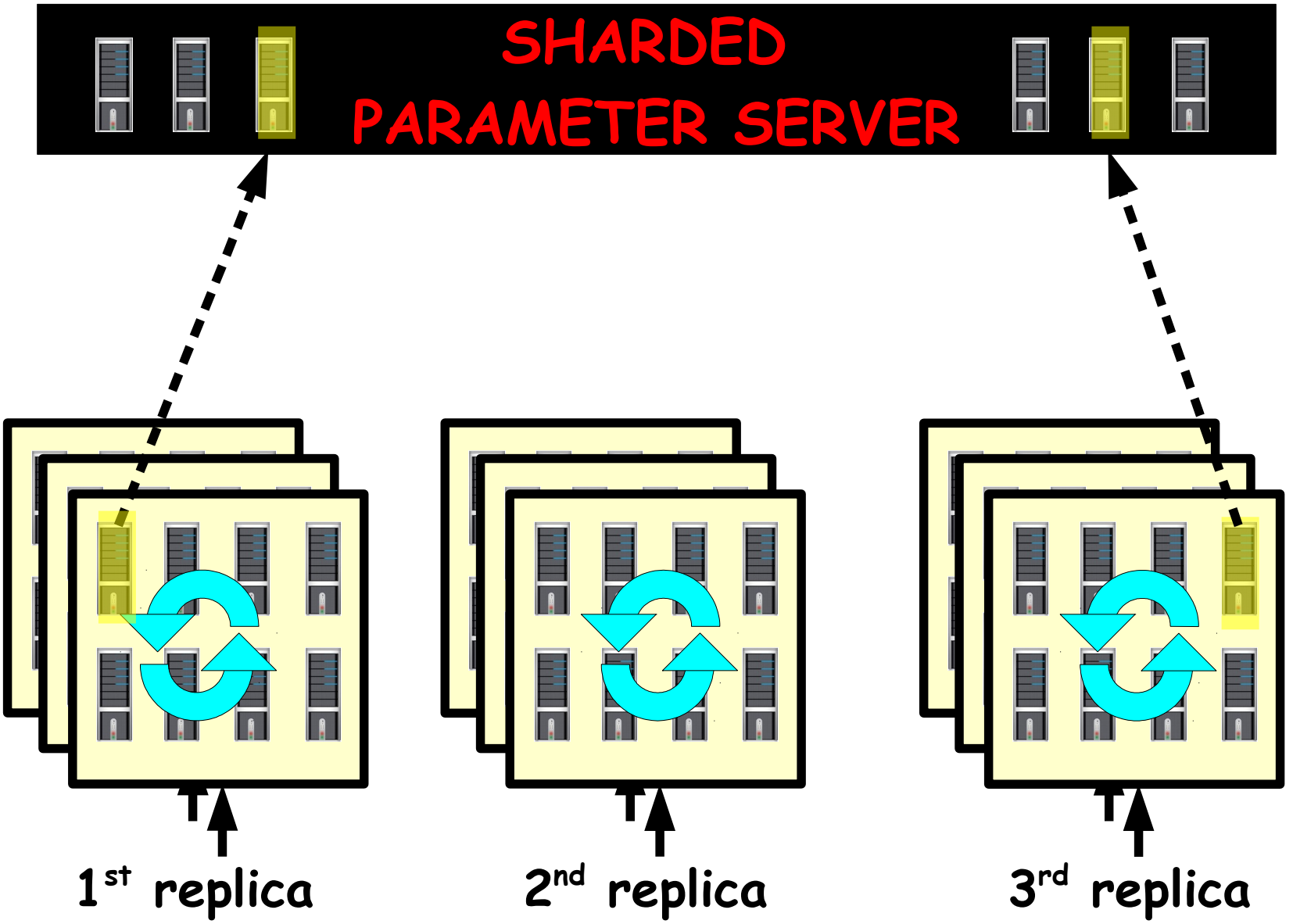


Asynchronous SGD

PARAMETER SERVER
(update parameters)

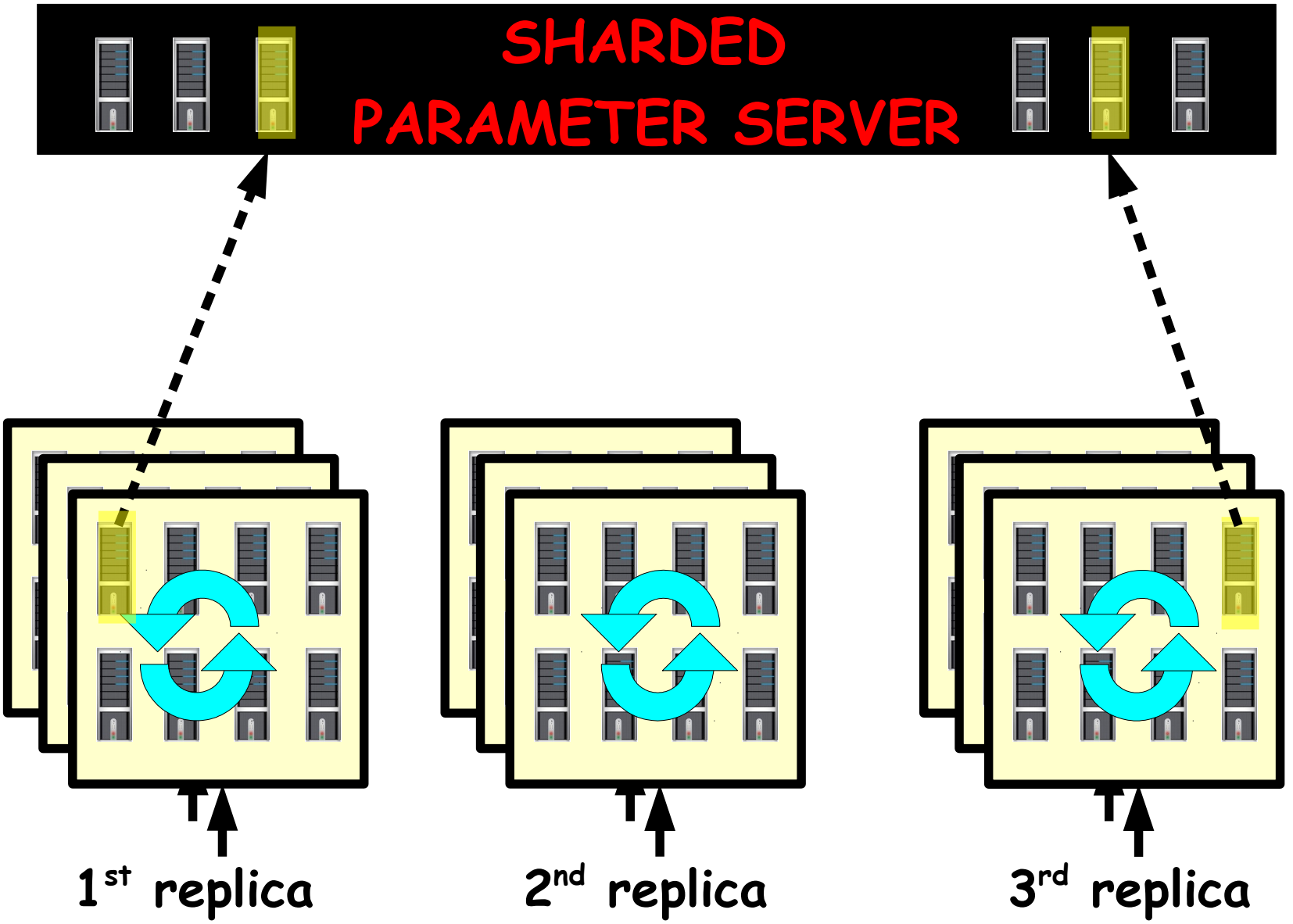


Highly Distributed Asynchronous SGD



Problem: each parameter needs its own learning rate!

Highly Distributed Asynchronous SGD



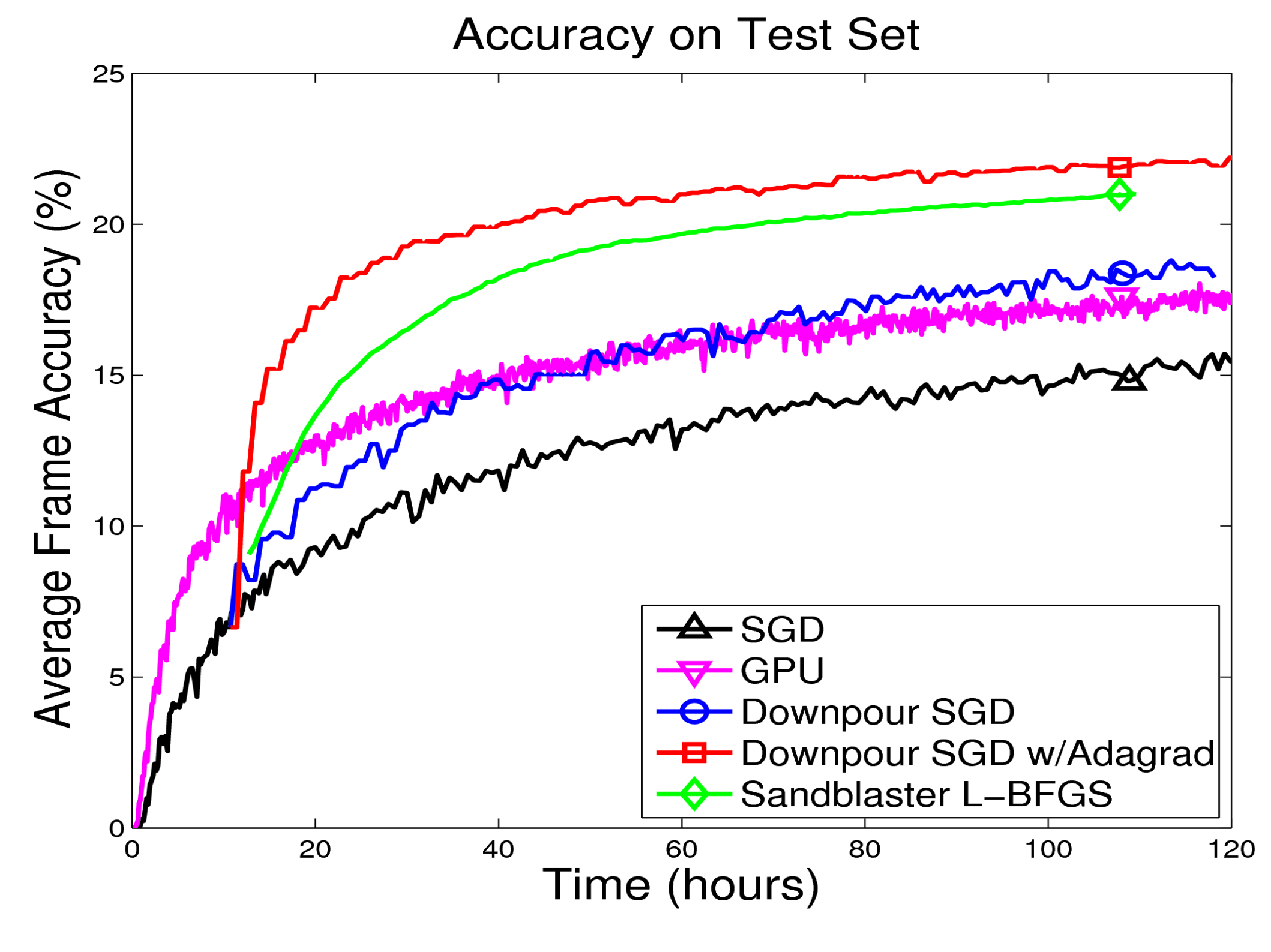
Adagrad

$$\theta_t^i \leftarrow \theta_{t-1}^i - \eta_t^i \frac{\partial L}{\partial \theta_t^i}$$

$$\eta_t^i = \frac{\bar{\eta}}{\sqrt{\sum_{k=0}^{t} \left(\frac{\partial L}{\partial \theta_k^i}\right)^2}}$$

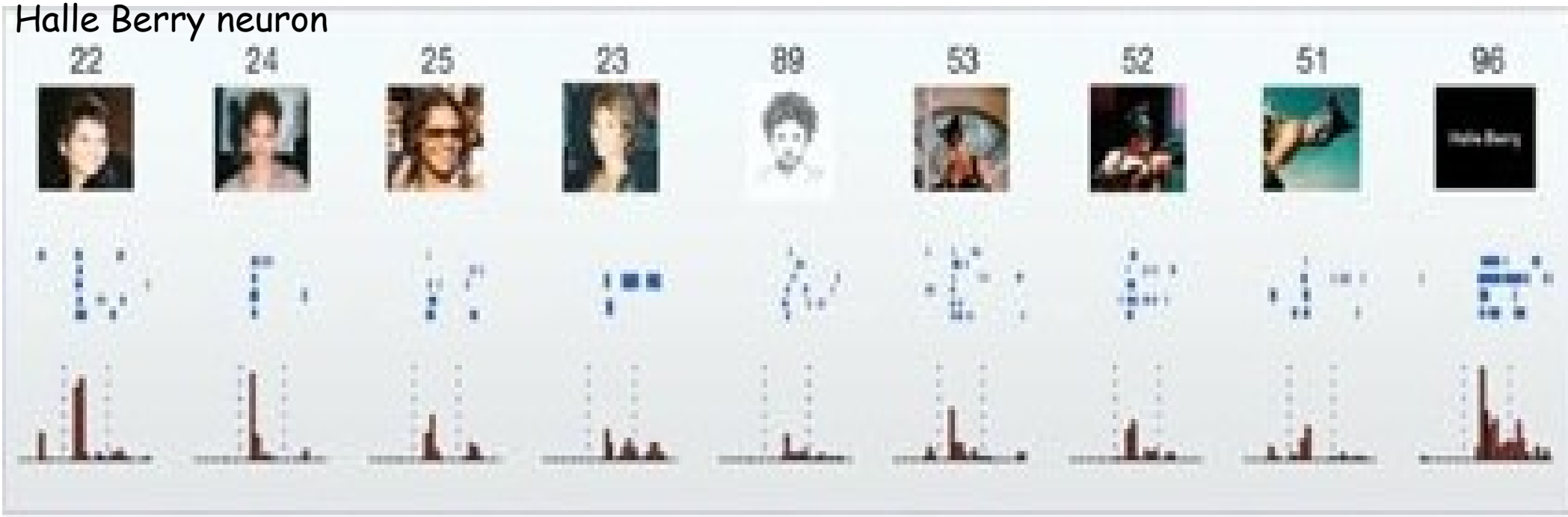
- similar to diagonal approx. of Hessian
- takes care of different scaling of learning rates
- handles annealing automatically

Duchi et al. "Adaptive subgradient methods for online learning and stochastic optimization" JMLR 2011


Accuracy on Test Set
Average Frame Accuracy (%)
Time (hours)
0
5
10
15
20
25
0
20
40
60
80
100
120
SGD
GPU
Downpour SGD
Downpour SGD w/Adagrad
Sandblaster L–BFGS

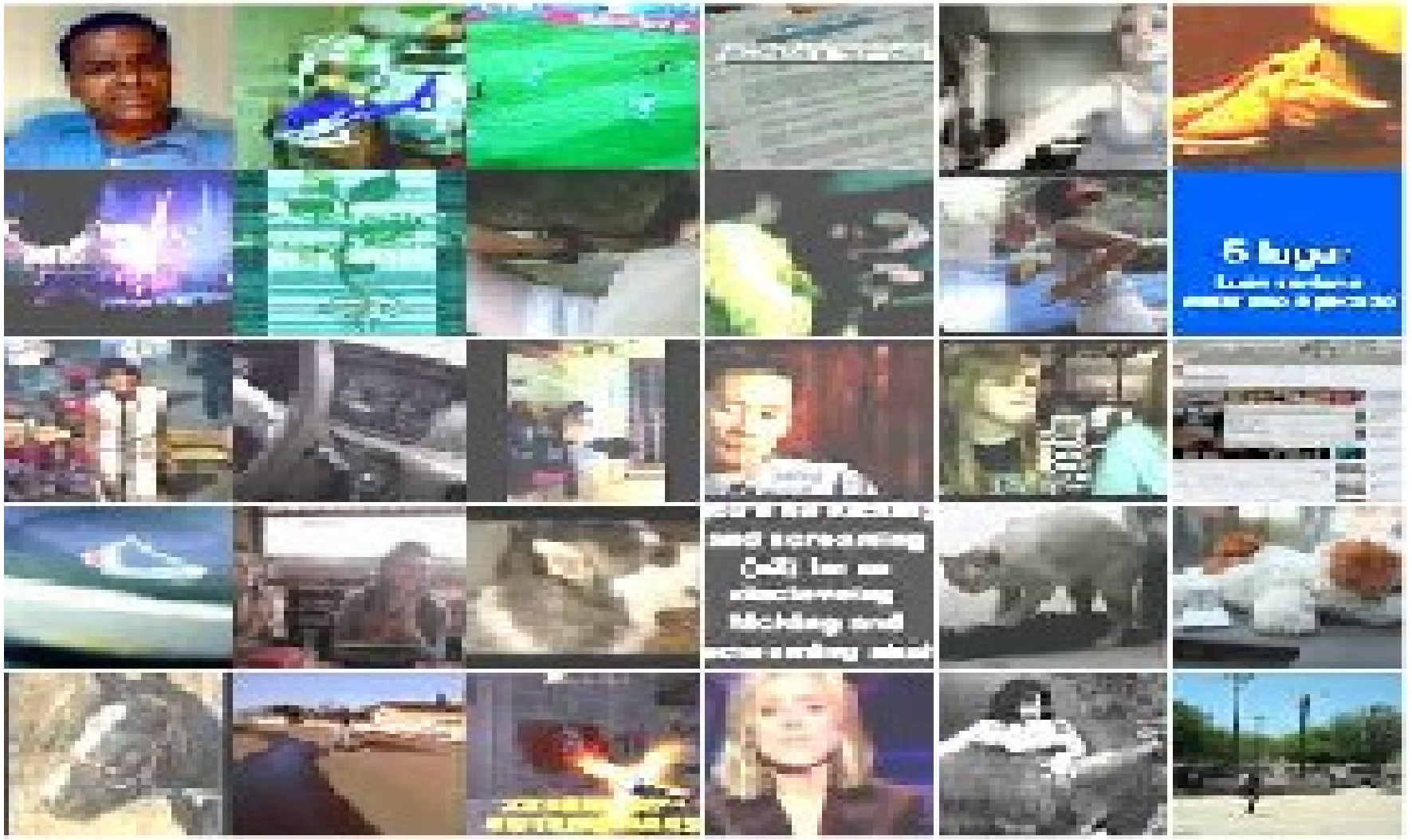
from Quiroga et al. "Invariant visual representation by single neurons in the human brain" Nature 2005

"Here we report on a remarkable subset of MTL neurons that are selectively activated by strikingly different pictures of given individuals, landmarks or objects and in some cases even by letter strings with their names."



Unsupervised Learning With 1B Parameters

DATA: 10M youtube (unlabeled) frames of size 200x200.



Le et al. "Building high-level features using large-scale unsupervised learning" ICML 2012

Unsupervised Learning With 1B Parameters

Deep Net:

- 3 stages
- each stage consists of local filtering, L2 pooling, LCN
  - 18x18 filters
  - 8 filters at each location
  - L2 pooling and LCN over 5x5 neighborhoods
- training jointly the three layers by:
  - reconstructing the input of each layer
  - sparsity on the code

Le et al. "Building high-level features using large-scale unsupervised learning" ICML 2012

Unsupervised Learning With 1B Parameters

Deep Net:

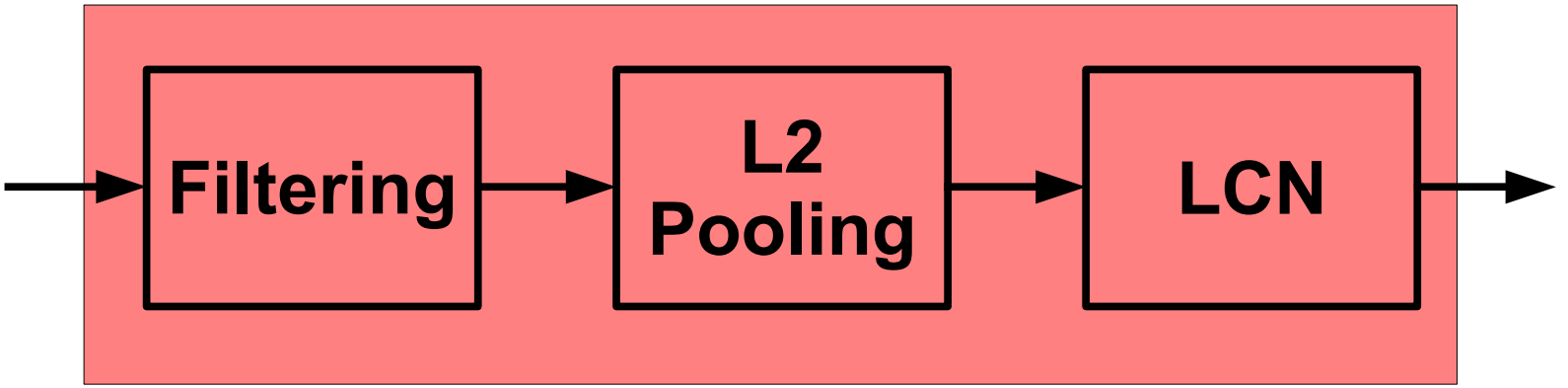
- 3 stages
- each stage consists of local filtering, L2 pooling, LCN
  - 18x18 filters
  - 8 filters at each location
  - L2 pooling and LCN over 5x5 neighborhoods
- training jointly the three layers by:
  - reconstructing the input of each layer
  - sparsity on the code

1B parameters!!!

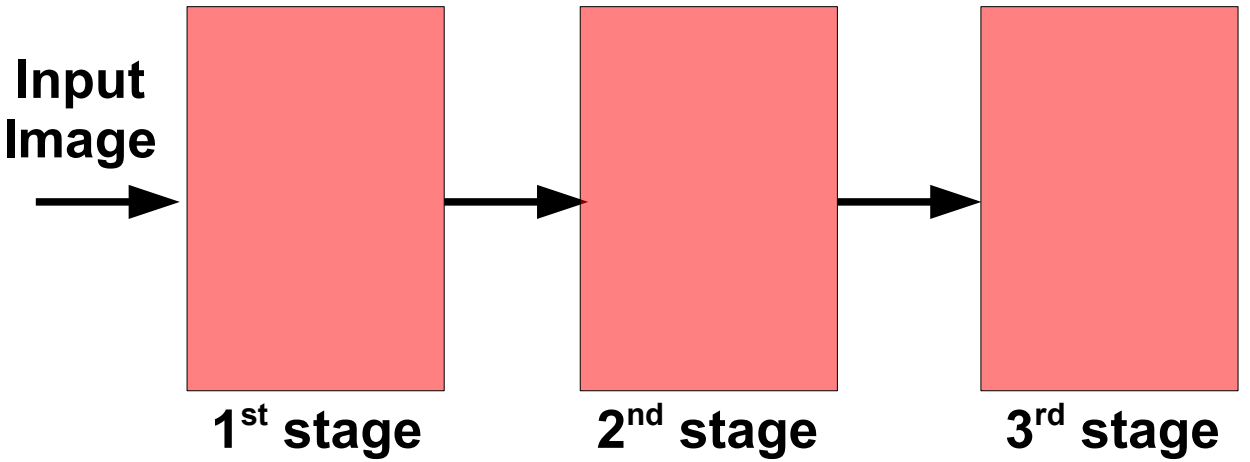
Le et al. "Building high-level features using large-scale unsupervised learning" ICML 2012

Unsupervised Learning With 1B Parameters

One stage (zoom)



Whole system



Unsupervised Learning With 1B Parameters

$$L = ReconstructionErrorLayer(1) + ReconstructionErrorLayer(2) + ReconstructionErrorLayer(3) + \lambda\, SparsityLayer(3)$$

Validating Unsupervised Learning

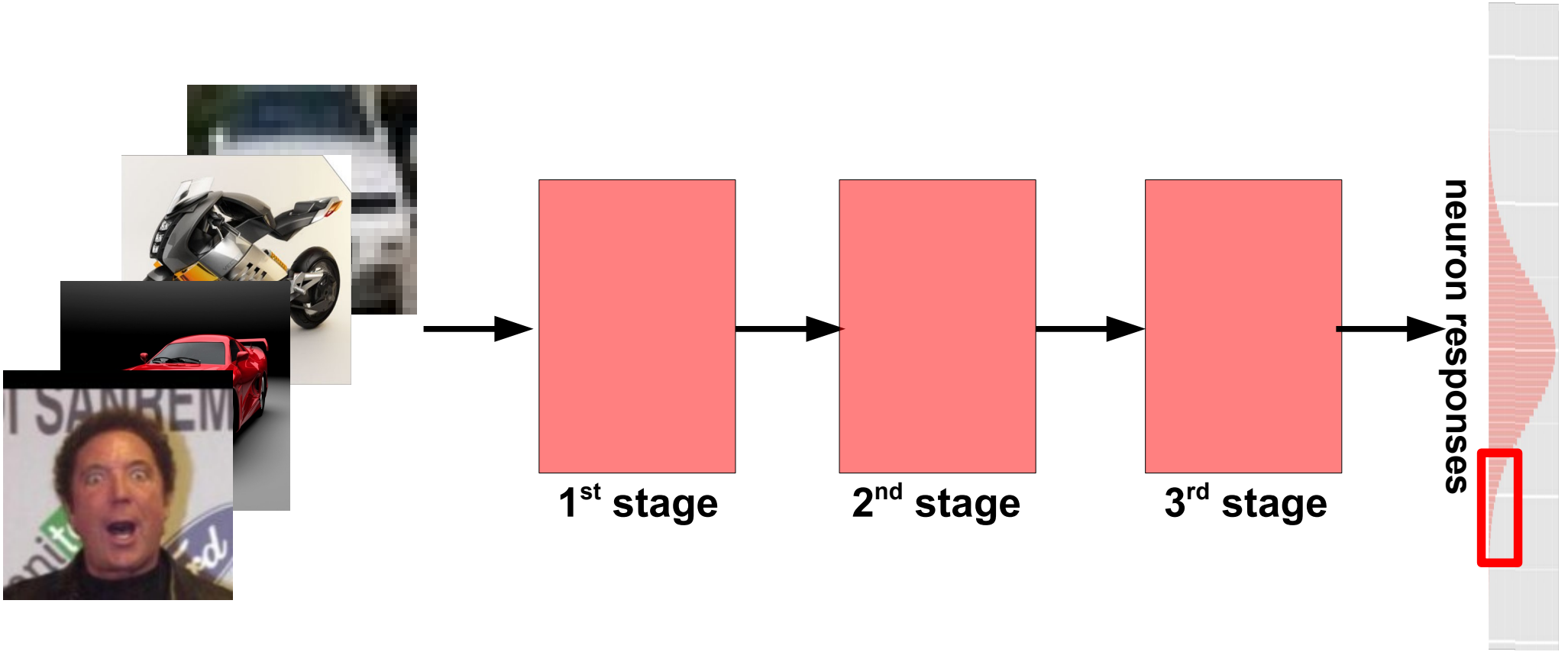
The network has seen lots of objects during training, but without any label.

Q.: how can we validate unsupervised learning?

Q.: Did the network form any high-level representation?
E.g., does it have any neuron responding for faces?

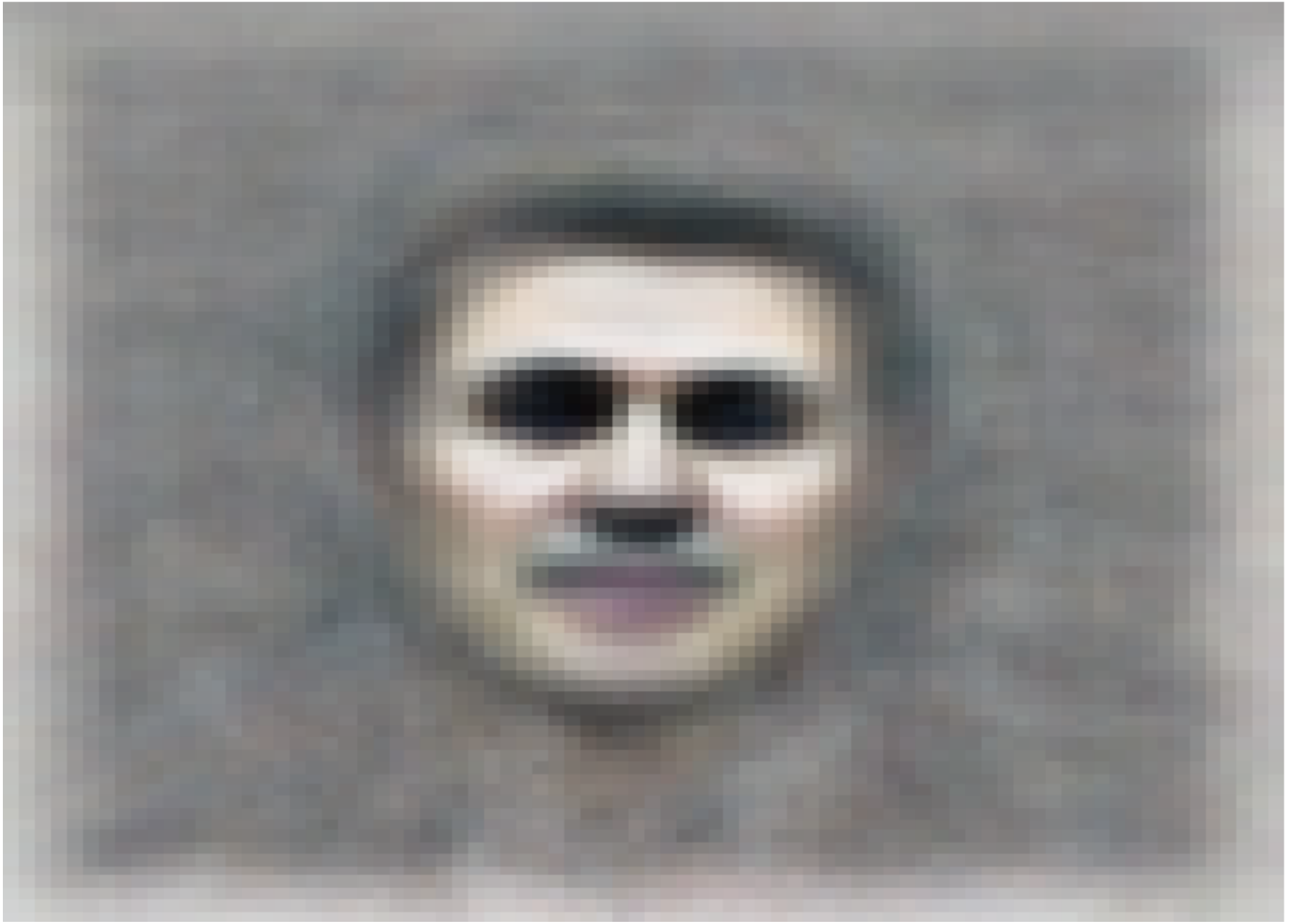
- build validation set with 50% faces, 50% random images
- study properties of neurons

Validating Unsupervised Learning

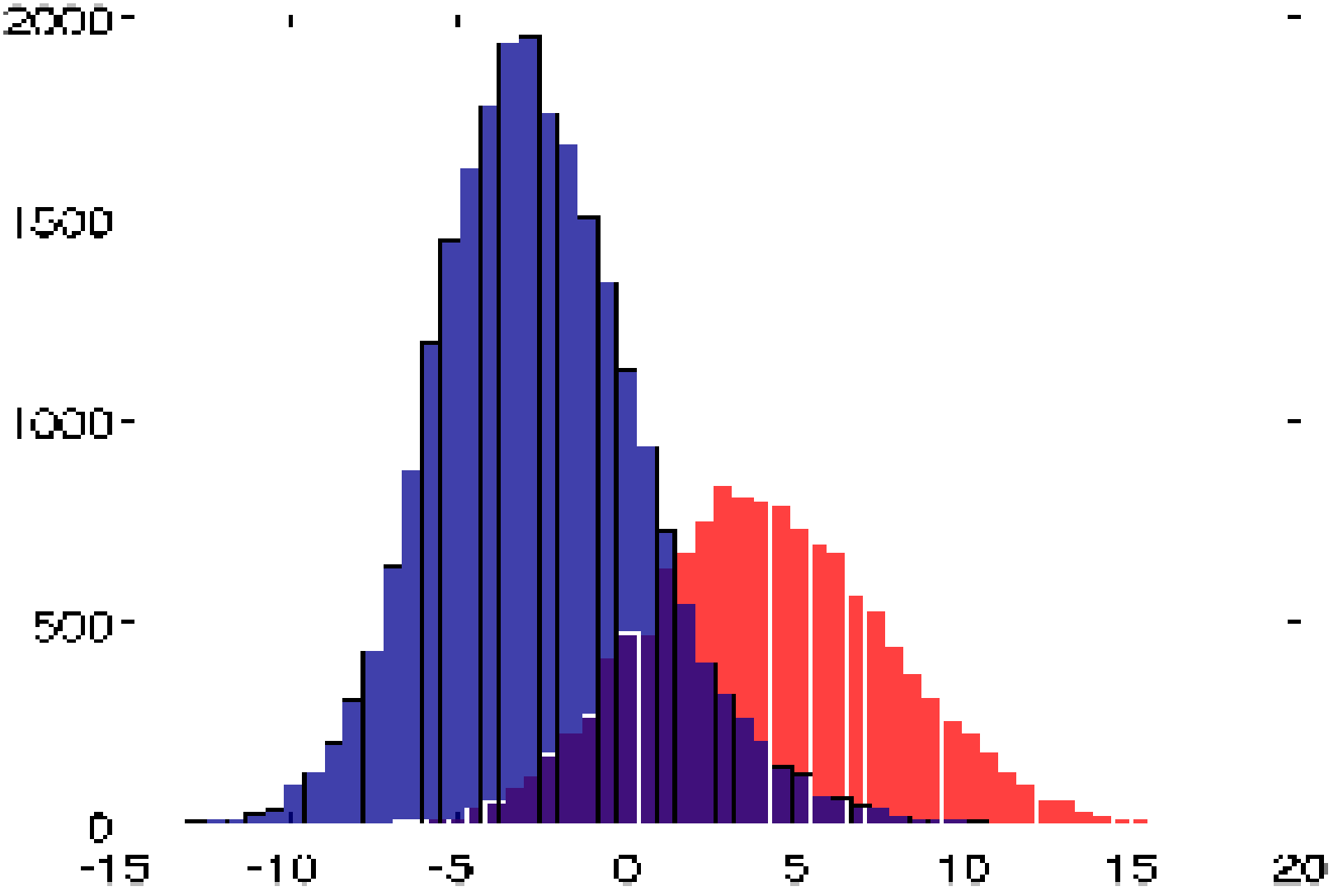


Top Images For Best Face Neuron

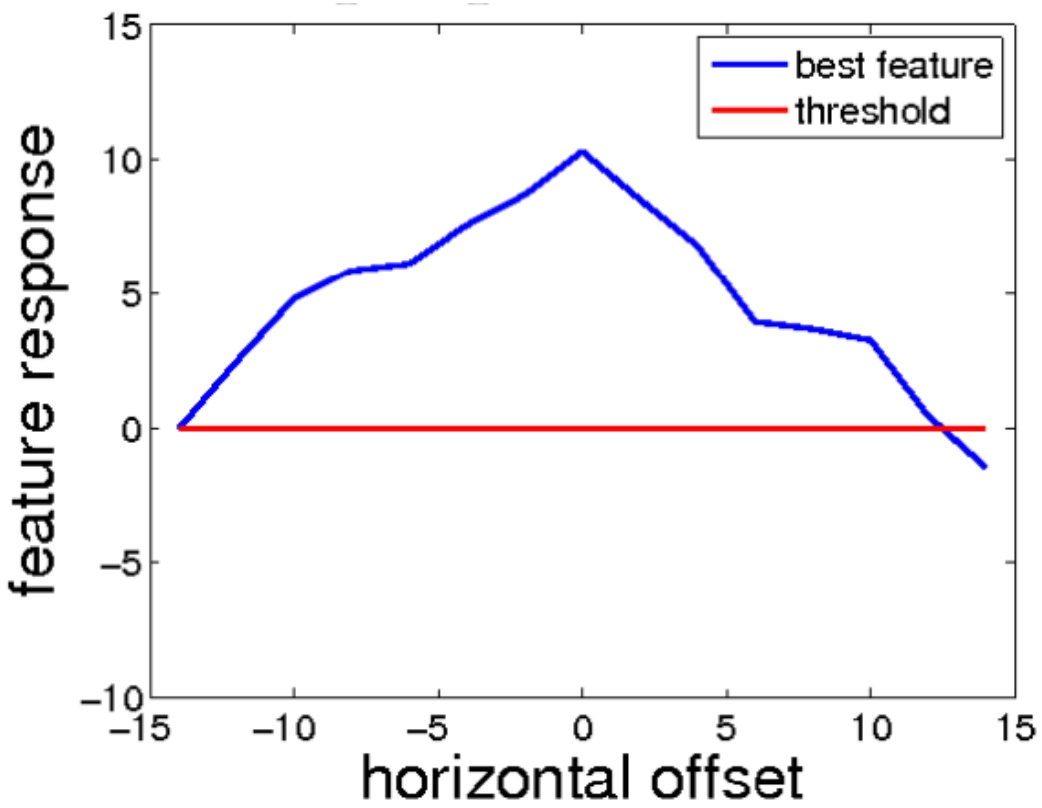
Best Input For Face Neuron

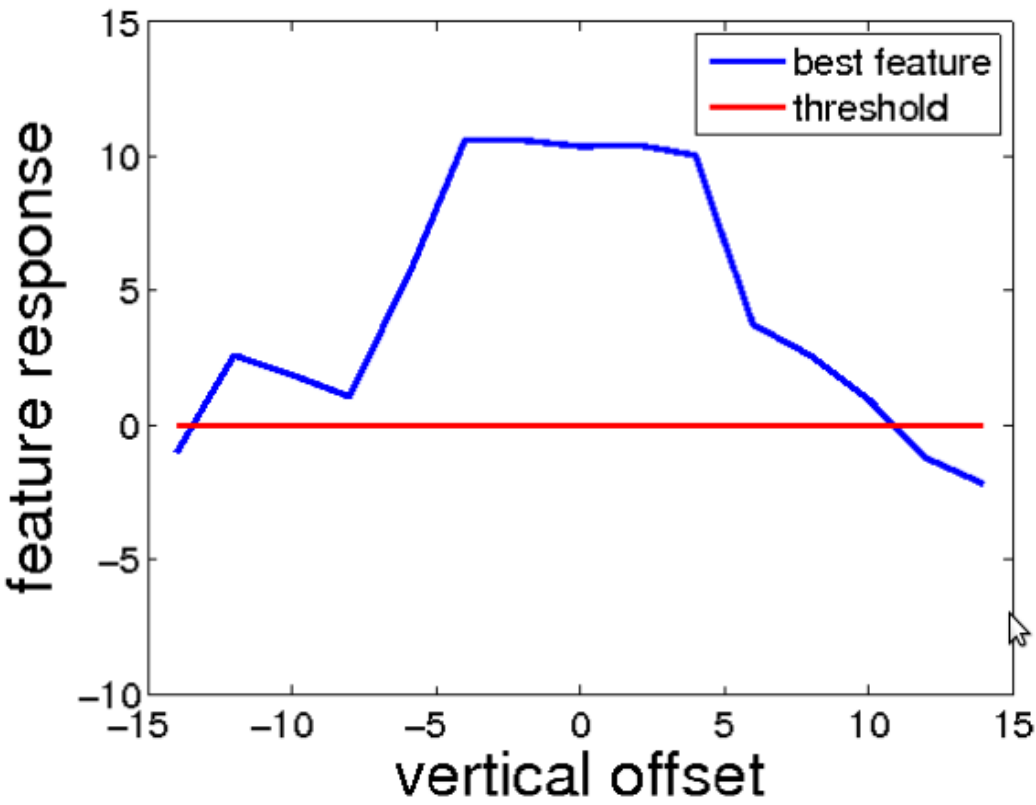


Face / No face

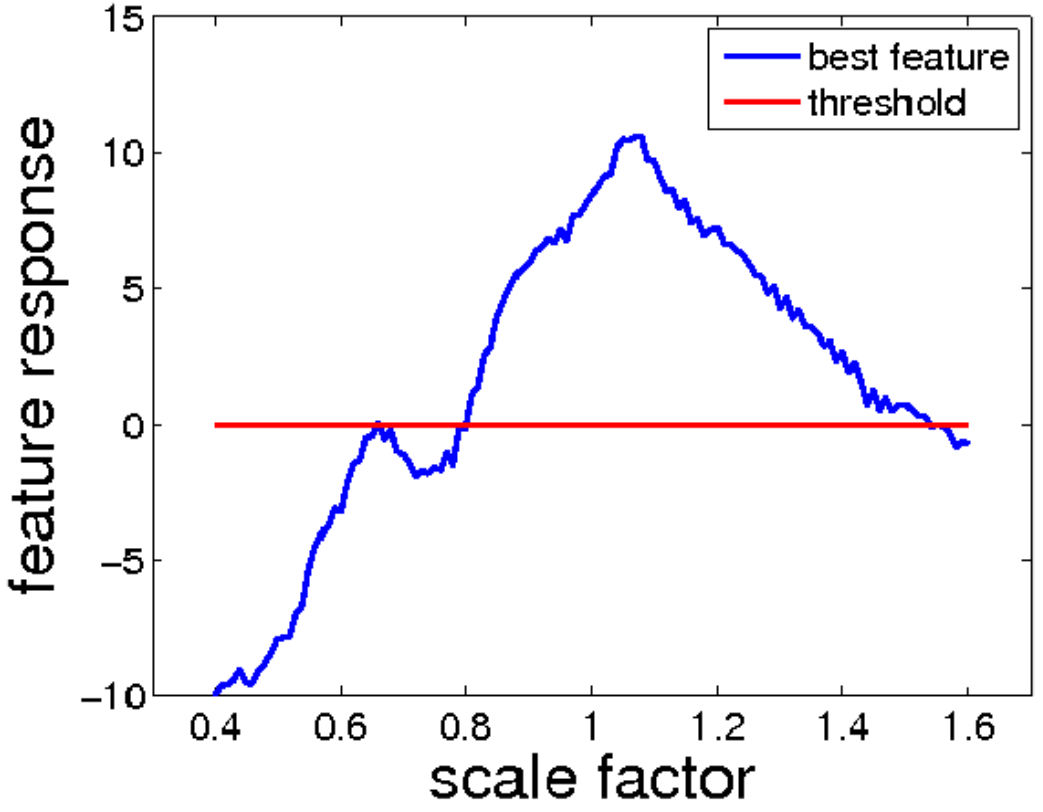


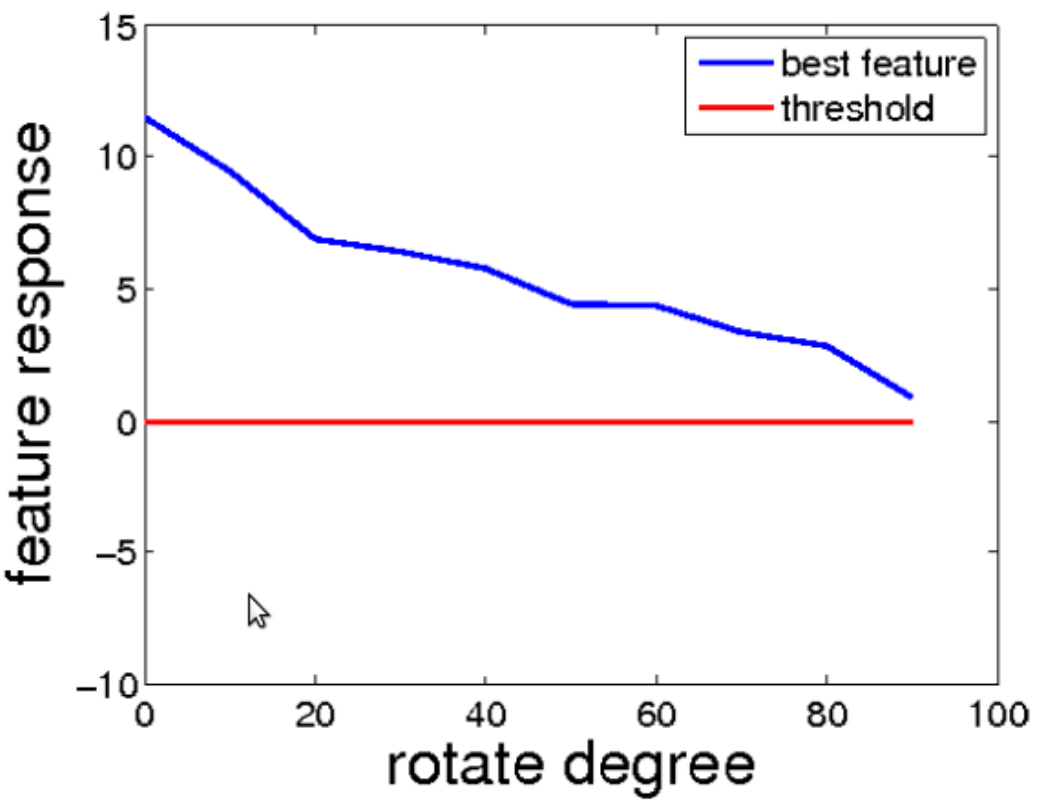
Invariance Properties





Invariance Properties





Comparison: Face Neuron

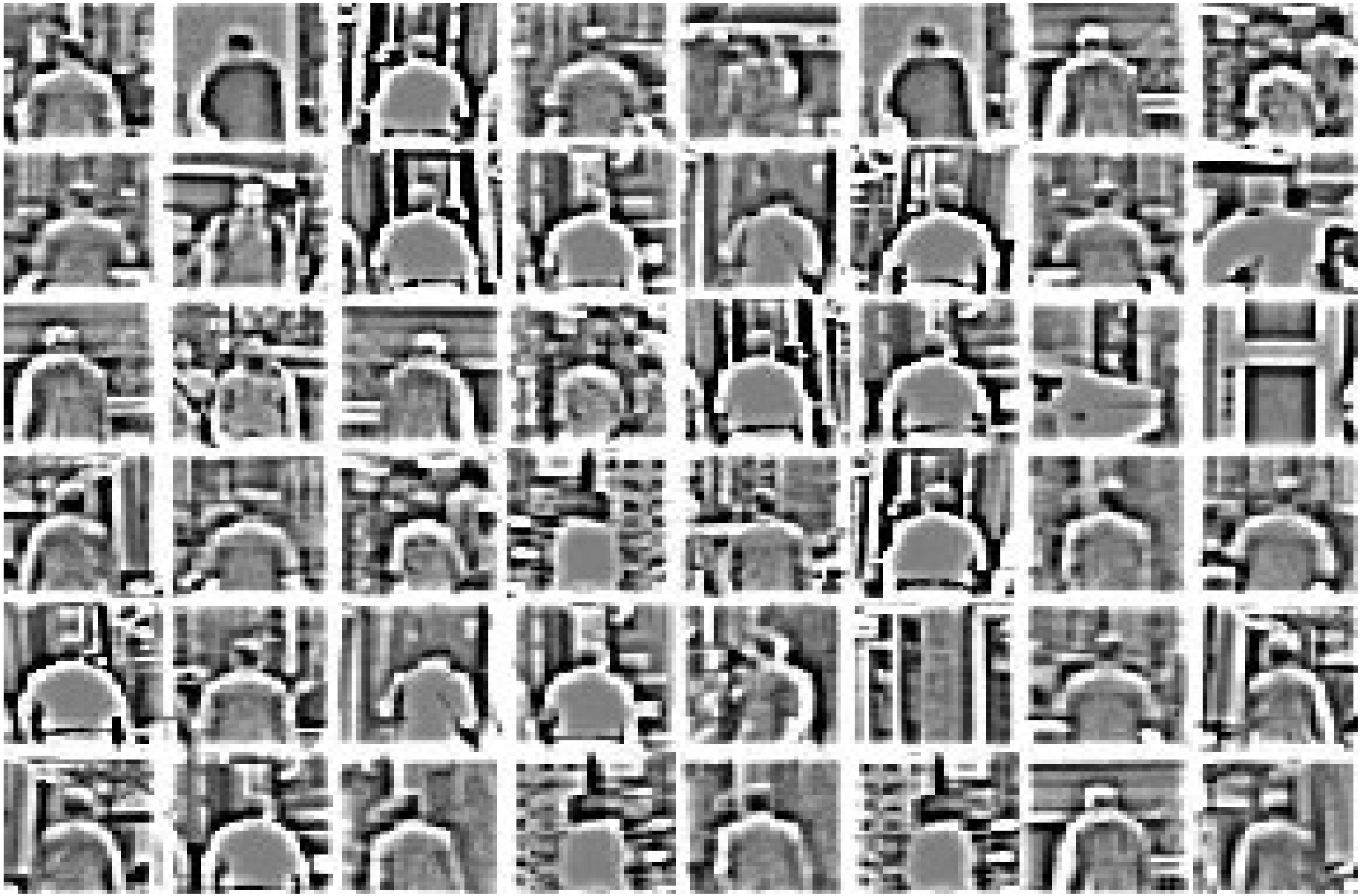
Method	Accuracy %
Random guess	65
Best training sample	74
1 layer net	71
Deep net before training	67
Deep net after training	81
Deep net after training without LCN	78

Comparison: Face Neuron

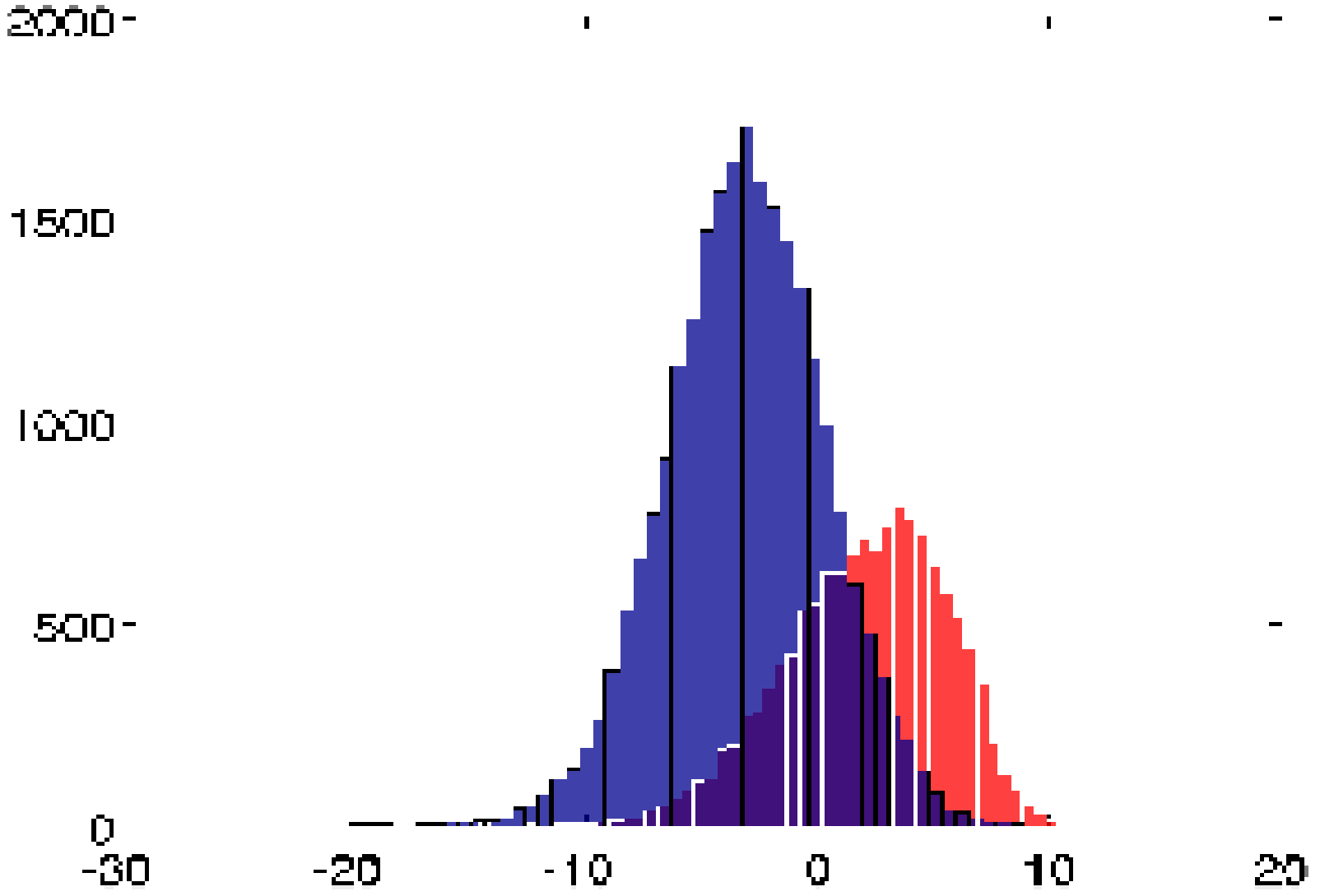
Method	Accuracy %
K-Means on 40x40, k=30000	72
3-layer Autoencoder	72
Deep net after training	81

Pedestrian Neuron

Top Images for Pedestrian Neuron



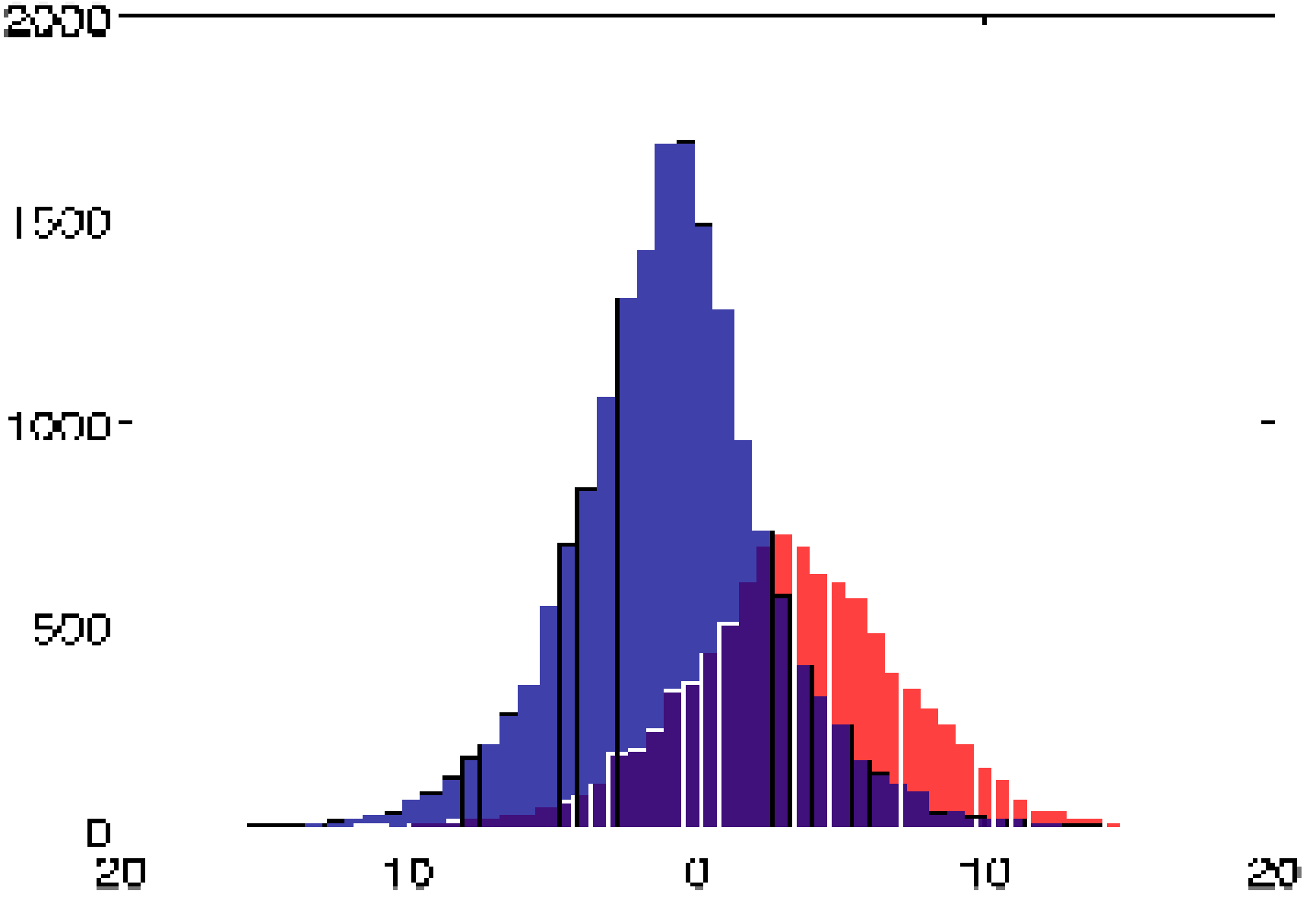
Pedestrian Neuron



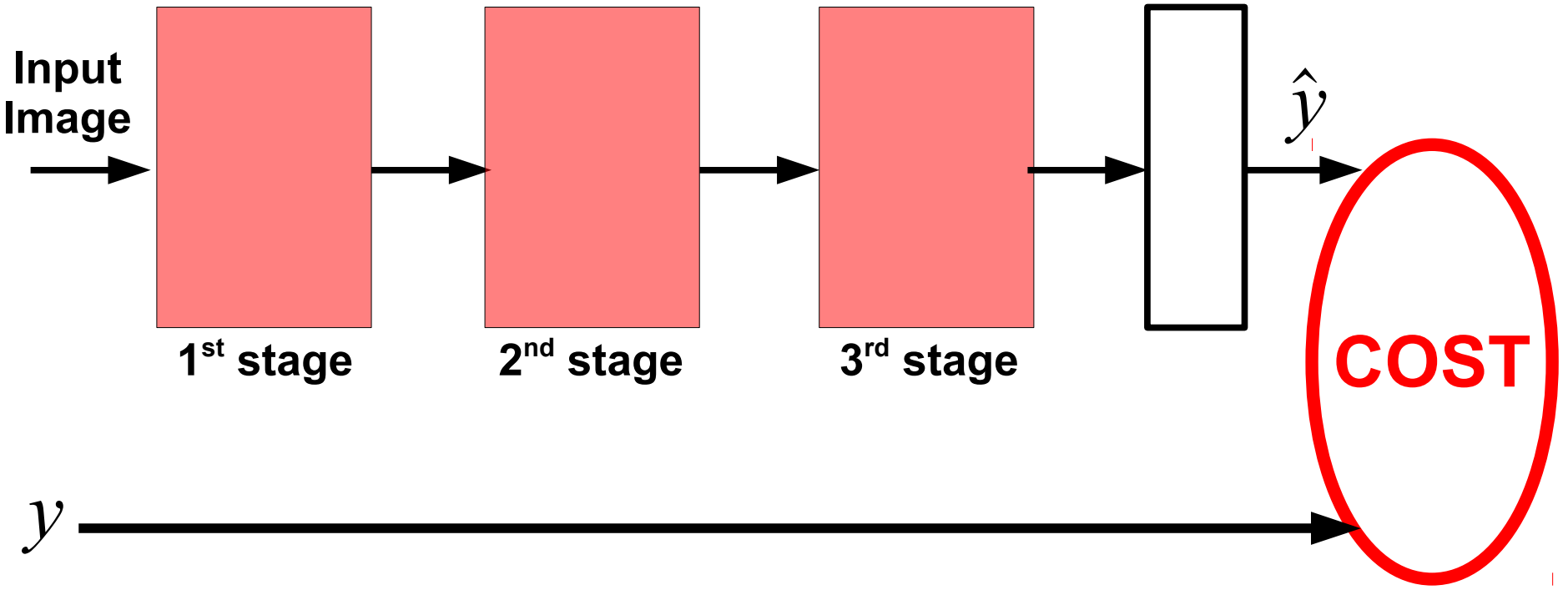
Cat Neuron

Top Images for Cat Neuron

Cat Neuron



Unsupervised + Supervised (ImageNet)



Object Recognition on ImageNet

IMAGENET v.2011 (16M images, 20K categories)

METHOD	ACCURACY %
Weston & Bengio 2011	9.3
Linear Classifier on deep features	13.1
Deep Net (from random)	13.6
Deep Net (from unsup.)	15.8

Top Inputs After Supervision

Top Inputs After Supervision

References on ConvNets & alike

Tutorials & Background Material

- Yoshua Bengio, Learning Deep Architectures for AI, Foundations and Trends in Machine Learning, 2(1), pp.1-127, 2009.

- LeCun, Chopra, Hadsell, Ranzato, Huang: A Tutorial on Energy-Based Learning, in Bakir, G. and Hofman, T. and Schölkopf, B. and Smola, A. and Taskar, B. (Eds), Predicting Structured Data, MIT Press, 2006

Convolutional Nets

- LeCun, Bottou, Bengio and Haffner: Gradient-Based Learning Applied to Document Recognition, Proceedings of the IEEE, 86(11):2278-2324, November 1998

- Jarrett, Kavukcuoglu, Ranzato, LeCun: What is the Best Multi-Stage Architecture for Object Recognition?, Proc. International Conference on Computer Vision (ICCV'09), IEEE, 2009

- Kavukcuoglu, Sermanet, Boureau, Gregor, Mathieu, LeCun: Learning Convolutional Feature Hierachies for Visual Recognition, Advances in Neural Information Processing Systems (NIPS 2010), 23, 2010

Unsupervised Learning

- ICA with Reconstruction Cost for Efficient Overcomplete Feature Learning. Le, Karpenko, Ngiam, Ng. In NIPS*2011

- Rifai, Vincent, Muller, Glorot, Bengio, Contracting Auto-Encoders: Explicit invariance during feature extraction, in: Proceedings of the Twenty-eight International Conference on Machine Learning (ICML'11), 2011

- Vincent, Larochelle, Lajoie, Bengio, Manzagol, Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion, Journal of Machine Learning Research, 11:3371--3408, 2010.

- Gregor, Szlam, LeCun: Structured Sparse Coding via Lateral Inhibition, Advances in Neural Information Processing Systems (NIPS 2011), 24, 2011

- Kavukcuoglu, Ranzato, LeCun. "Fast Inference in Sparse Coding Algorithms with Applications to Object Recognition". ArXiv 1010.3467 2008

- Hinton, Krizhevsky, Wang, Transforming Auto-encoders, ICANN, 2011

Multi-modal Learning

- Multimodal deep learning, Ngiam, Khosla, Kim, Nam, Lee, Ng. In Proceedings of the Twenty-Eighth International Conference on Machine Learning, 2011.

Locally Connected Nets

- Gregor, LeCun "Emergence of complex-like cells in a temporal product network with local receptive fields" Arxiv. 2009
- Ranzato, Mnih, Hinton "Generating more realistic images using gated MRF's" NIPS 2010
- Le, Ngiam, Chen, Chia, Koh, Ng "Tiled convolutional neural networks" NIPS 2010

Distributed Learning

- Le, Ranzato, Monga, Devin, Corrado, Chen, Dean, Ng. "Building High-Level Features Using Large Scale Unsupervised Learning". International Conference of Machine Learning (ICML 2012), Edinburgh, 2012.

Papers on Scene Parsing

- Farabet, Couprie, Najman, LeCun, "Scene Parsing with Multiscale Feature Learning, Purity Trees, and Optimal Covers", in Proc. of the International Conference on Machine Learning (ICML'12), Edinburgh, Scotland, 2012.
- Socher, Lin, Ng, Manning, "Parsing Natural Scenes and Natural Language with Recursive Neural Networks". International Conference of Machine Learning (ICML 2011) 2011.

Papers on Object Recognition

- Boureau, Le Roux, Bach, Ponce, LeCun: Ask the locals: multi-way local pooling for image recognition, Proc. International Conference on Computer Vision 2011

- Sermanet, LeCun: Traffic Sign Recognition with Multi-Scale Convolutional Networks, Proceedings of International Joint Conference on Neural Networks (IJCNN'11)

- Ciresan, Meier, Gambardella, Schmidhuber. Convolutional Neural Network Committees For Handwritten Character Classification. 11th International Conference on Document Analysis and Recognition (ICDAR 2011), Beijing, China.

- Ciresan, Meier, Masci, Gambardella, Schmidhuber. Flexible, High Performance Convolutional Neural Networks for Image Classification. International Joint Conference on Artificial Intelligence IJCAI-2011.

Papers on Action Recognition

- Learning hierarchical spatio-temporal features for action recognition with independent subspace analysis, Le, Zou, Yeung, Ng. In Computer Vision and Pattern Recognition (CVPR), 2011

Papers on Segmentation

- Turaga, Briggman, Helmstaedter, Denk, Seung Maximin learning of image segmentation. NIPS, 2009.

Papers on Vision for Robotics

- Hadsell, Sermanet, Scoffier, Erkan, Kavackuoglu, Muller, LeCun: Learning Long-Range Vision for Autonomous Off-Road Driving, Journal of Field Robotics, 26(2):120-144, February 2009,

Deep Convex Nets & Deconv-Nets

- Deng, Yu. "Deep Convex Network: A Scalable Architecture for Speech Pattern Classification." Interspeech, 2011.

- Zeiler, Taylor, Fergus "Adaptive Deconvolutional Networks for Mid and High Level Feature Learning." ICCV. 2011

Papers on Biological Inspired Vision

- Serre, Wolf, Bileschi, Riesenhuber, Poggio. Robust Object Recognition with Cortex-like Mechanisms, IEEE Transactions on Pattern Analysis and Machine Intelligence, 29, 3, 411-426, 2007.

- Pinto, Doukhan, DiCarlo, Cox "A high-throughput screening approach to discovering good forms of biologically inspired visual representation." {PLoS} Computational Biology. 2009

Papers on Embedded ConvNets for Real-Time Vision Applications

- Farabet, Martini, Corda, Akselrod, Culurciello, LeCun: NeuFlow: A Runtime Reconfigurable Dataflow Processor for Vision, Workshop on Embedded Computer Vision, CVPR 2011,

Papers on Image Denoising Using Neural Nets

- Burger, Schuler, Harmeling: Image Denoisng: Can Plain Neural Networks Compete with BM3D?, Computer Vision and Pattern Recognition, CVPR 2012,

ConvNets & Invariance from a Mathematical Perspective

- Bruna, Mallat: Invariance Scattering Convolutional Network, PAMI 2012,

ConvNets to Learn Embeddings

- Hadsell, Chopra, LeCun: Dimensionality Reduction by Learning an Invariant Mapping, CVPR 2006,

Software & Links

Deep Learning website

- http://deeplearning.net/

C++ code for ConvNets

- http://eblearn.sourceforge.net/

Matlab code for R-ICA unsupervised algorithm

- http://ai.stanford.edu/~quocle/rica_release.zip

Python-based learning library

- http://deeplearning.net/software/theano/

Lush learning library which includes ConvNets

- http://lush.sourceforge.net/

Torch7: learning library that supports neural net training

http://www.torch.ch

Acknowledgements

Quoc Le, Andrew Ng

Google

Jeff Dean, Kai Chen, Greg Corrado, Matthieu Devin, Mark Mao, Rajat Monga, Paul Tucker, Samy Bengio

NYU

Yann LeCun, Pierre Sermanet, Clement Farabet