Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representations

Honglak Lee

Roger Grosse, Rajesh Ranganath, Andrew Ng

Computer Science Department

Stanford University

Outline

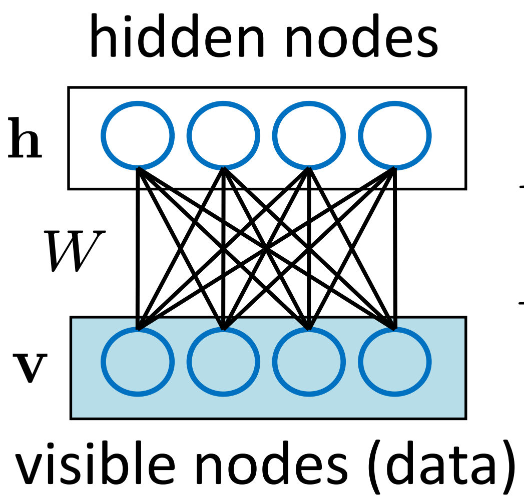
- Motivation
- Background
- Our Algorithms
- Experimental results
- Summary

Motivation

- "Deep" learning algorithms (Hinton et al., 2006; Bengio et al., 2006; Ranzato et al., 2007)
  - Inspired by hierarchical organization of the brain
  - Try to learn hierarchical feature representation where high level features are composed of simpler low level features
  - Mostly unsupervised
  - Single learning algorithm along the hierarchy
- We are interested in scaling up deep belief networks to learn generative models and to perform inference on challenging problems.

Background

- Restricted Boltzmann Machine (RBM)



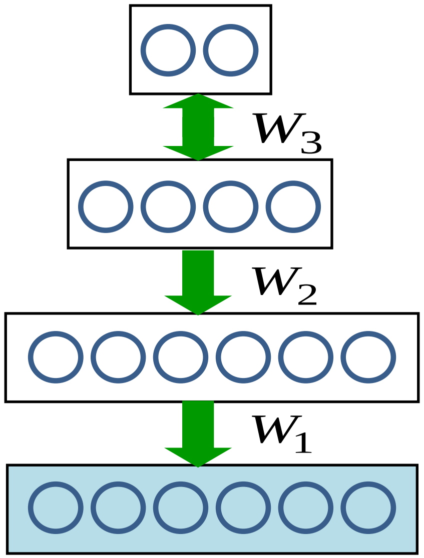
$$P(\mathbf{v},\mathbf{h}) = \frac{1}{Z}\exp(-E(\mathbf{v},\mathbf{h}))$$

$$E(\mathbf{v},\mathbf{h}) = -\sum_{i,j} v_i W_{i,j} h_j - \sum_j b_j h_j - \sum_i c_i v_i$$

  - Undirected, bipartite graphical model
  - Block Gibbs sampling is used for inference and learning
  - Unsupervised training using Contrastive Divergence approximation to maximum likelihood

Background

- Deep Belief Network (DBN) (Hinton et al., 2006)
  - Hierarchical generative model
  - Greedy layerwise training using Restricted Boltzmann machines
  - Applications
    - Recognizing handwritten digits
    - Learning motion capture data
  - Input Dimension ~ 1,000 (e.g., 30x30 pixels)



visible nodes (data)

- How can we scale to realistic image sizes (e.g. 200x200 pixels)?

Background

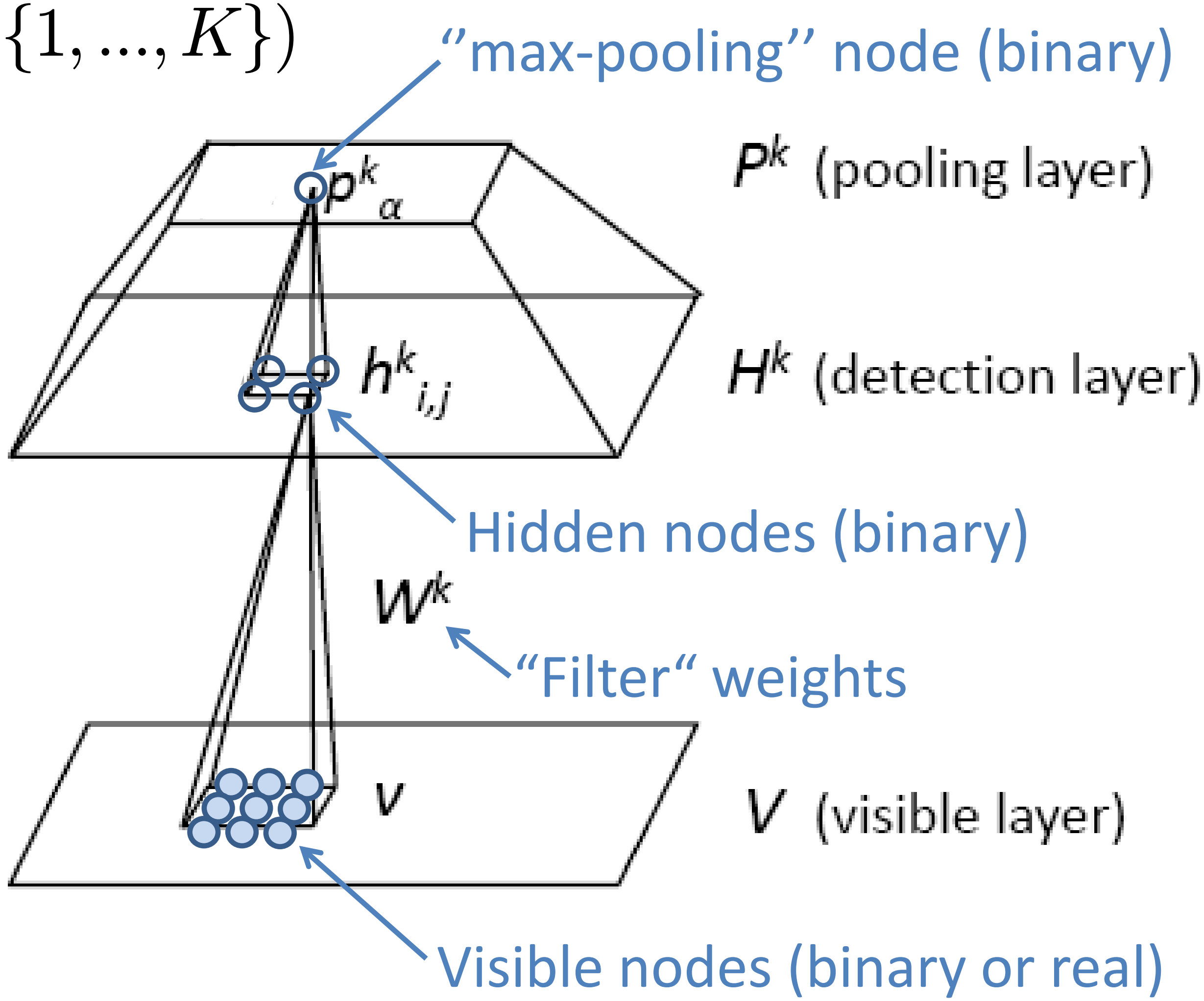
- Convolutional Architectures (e.g., LeCun et al., 1989)
  - Alternate between "detection" and "pooling" layers
  - Detection layers involve weights shared between all image locations; computed efficiently with convolution
  - Each pooling unit computes the maximum of the activation of several detection units.
    - Shrinks the representation in higher layers
    - Provides invariance to local transformations
- Max pooling is deterministic and feed-forward; we give it a *probabilistic semantics* that enables to *combine bottom-up and top-down information*.

Our Algorithms

Convolutional RBM (CRBM)

(Related work: Desjardins and Bengio, 2008)

For "filter" k,
$(k \in \{1, ..., K\})$



Convolutional RBM

- Joint Probability distribution

$$P(\mathbf{v},\mathbf{h}) = \frac{1}{Z}\exp(-E(\mathbf{v},\mathbf{h}))$$

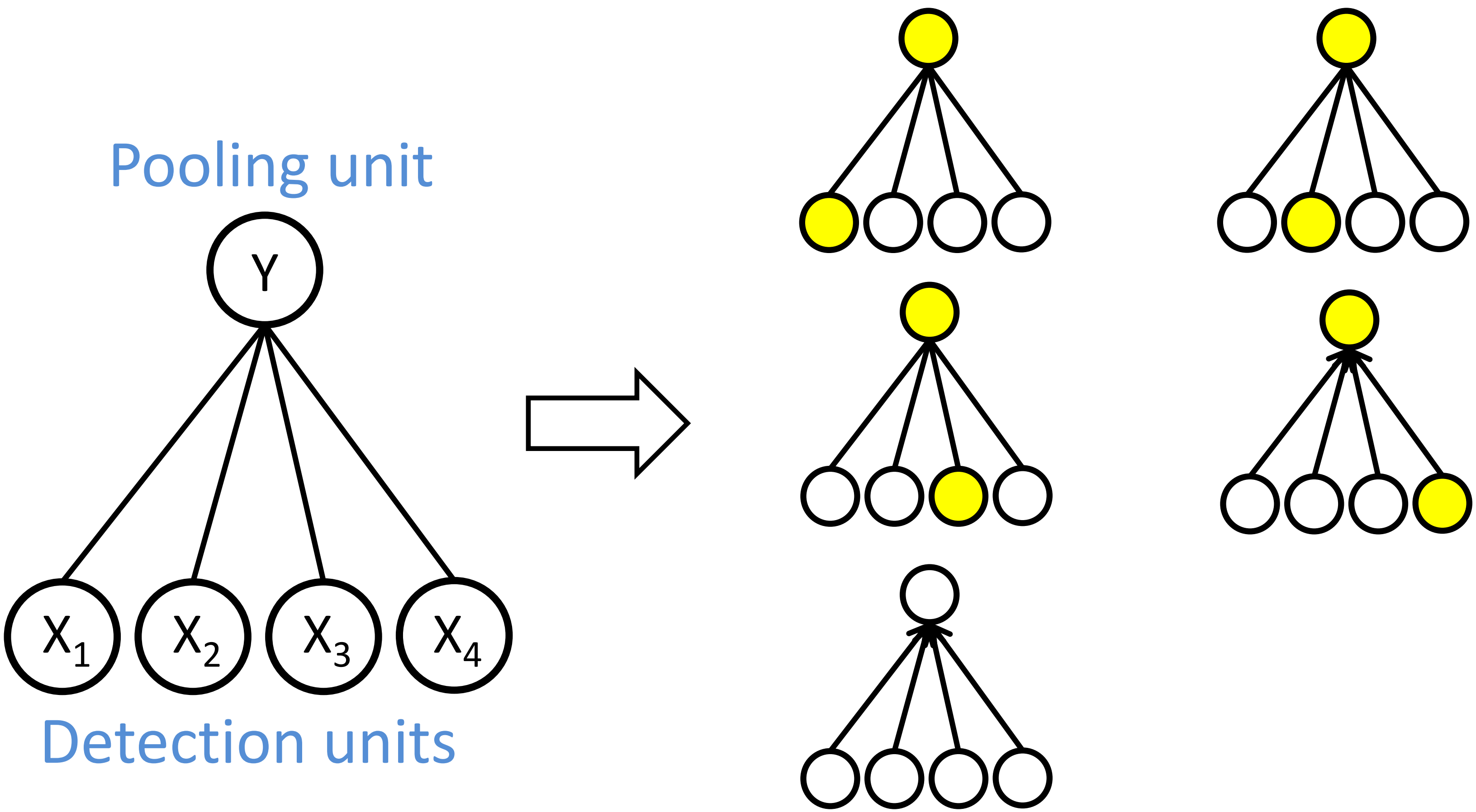
$$E(\mathbf{v},\mathbf{h}) = -\sum_k \sum_{i,j} \left(h^k_{i,j}(\tilde{W}^k * v)_{i,j} + b^k h^k_{i,j} \right) - c \sum_{i,j} v_{i,j}$$

convolution

$$\text{subject to} \quad \underbrace{\sum_{(i,j)\in B_\alpha} h^k_{i,j} \le 1, \forall k, \alpha.}_{\text{Constraint for } \textit{probabilistic max pooling}}$$

- Block Gibbs sampling using linear filtering followed by multinomial (softmax) sampling.
- Training using sparse RBM formulation (Lee et al., 2008)

Probabilistic Max pooling

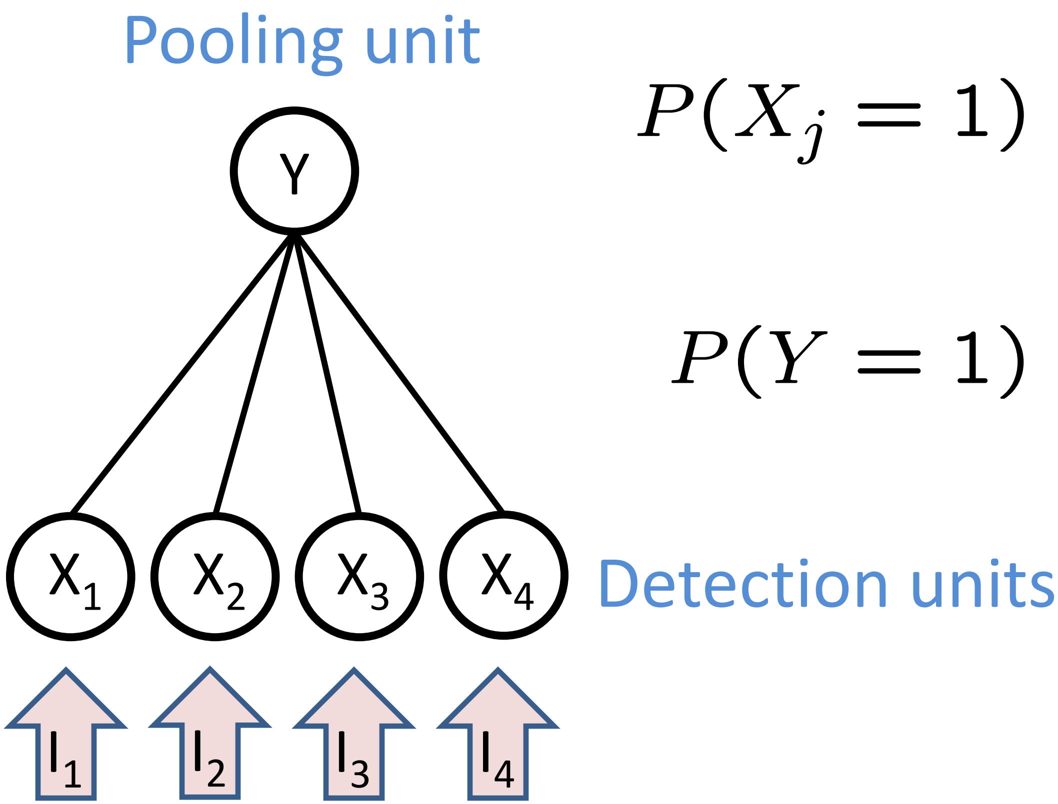


X_j are *stochastic binary* and *mutually exclusive*.

Collapse 2^n configurations into n+1 configurations. Permits bottom up and top down inference.

Probabilistic Max pooling

Bottom-up inference



$$P(X_j = 1) = \frac{\exp(I_j)}{1 + \sum_\ell \exp(I_\ell)}$$

$$P(Y = 1) = \frac{\sum_\ell \exp(I_\ell)}{1 + \sum_\ell \exp(I_\ell)}$$

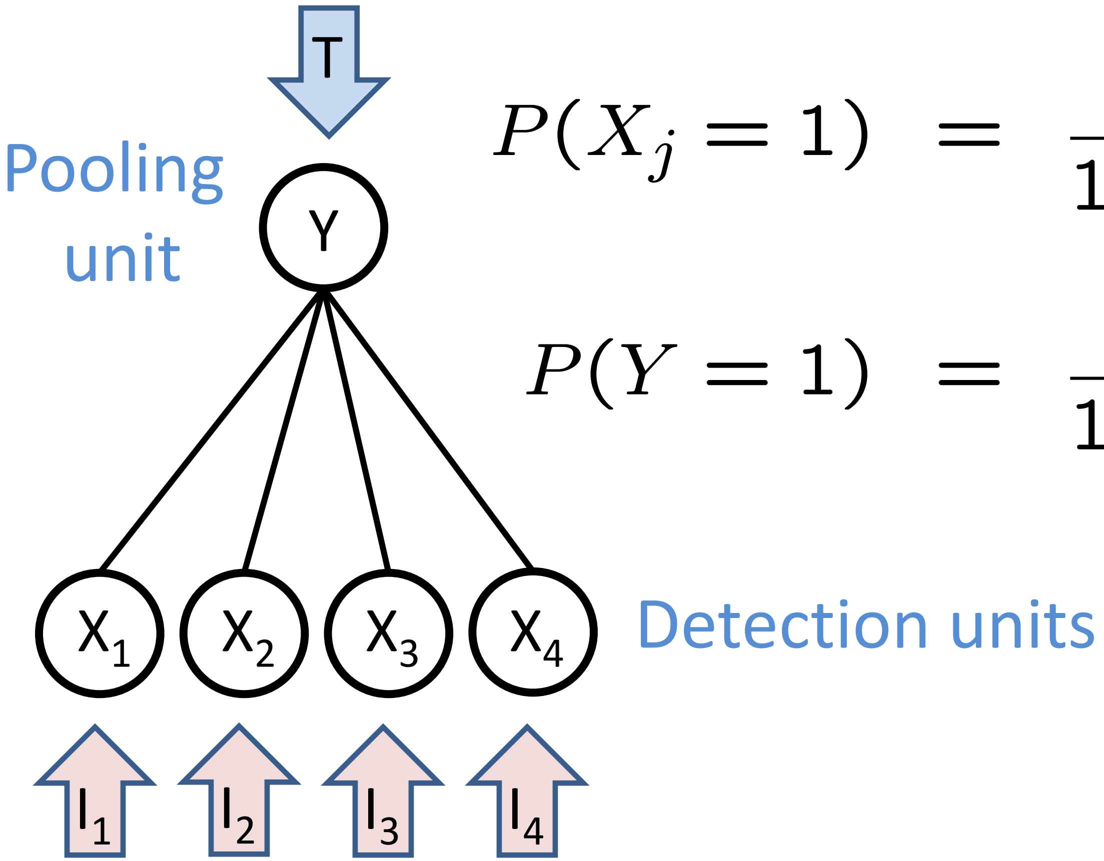
Convolutional Deep Belief Networks

- Greedy, layerwise Training
  - Train one layer (convolutional RBM) at a time.
    (Related work: Salakhutdinov and Hinton, 2009)

- Inference (approximate)
  - Undirected connections for all layers
  - Block Gibbs sampling or Mean-field
  - Hierarchical probabilistic inference

Hierarchical Probabilistic Inference

Combining bottom-up and top-down information



$$P(X_j = 1) = \frac{\exp(T + I_j)}{1 + \sum_\ell \exp(T + I_\ell)}$$

$$P(Y = 1) = \frac{\sum_\ell \exp(T + I_\ell)}{1 + \sum_\ell \exp(T + I_\ell)}$$

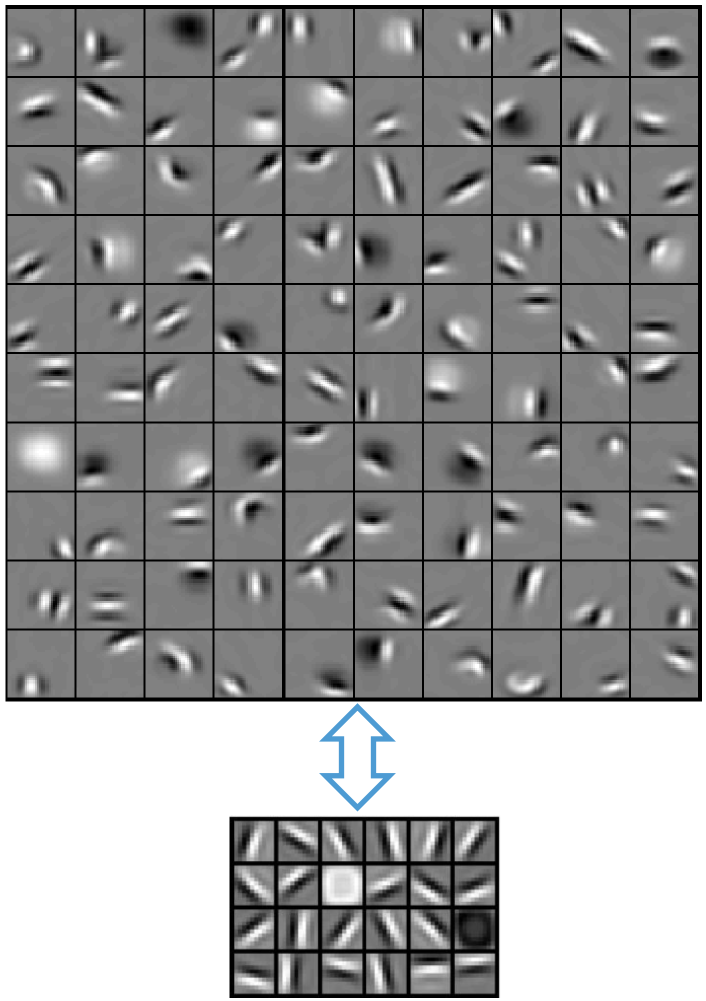
Experimental Results

Handwritten digit classification (MNIST)

- Trained a two-layer CDBN on unlabeled MNIST training data
- The first layer learns “strokes”; the second layer learns “groupings of the strokes.”
- Classification results (test error):

Labeled examples	1,000	2,000	3,000	5,000	60,000
CDBN	**2.62%**	**2.13%**	**1.91%**	**1.59%**	**0.82%**
Ranzato et al. (2007)	3.21%	2.53%	-	1.52%	0.64%
Hinton et al. (2006)	-	-	-	-	1.25%
Weston et al. (2008)	2.73%	-	1.83%	-	1.50%

Unsupervised learning from natural images



Second layer bases

Contours, Corners, Arcs, Surface boundaries

First layer bases

Localized, oriented edges

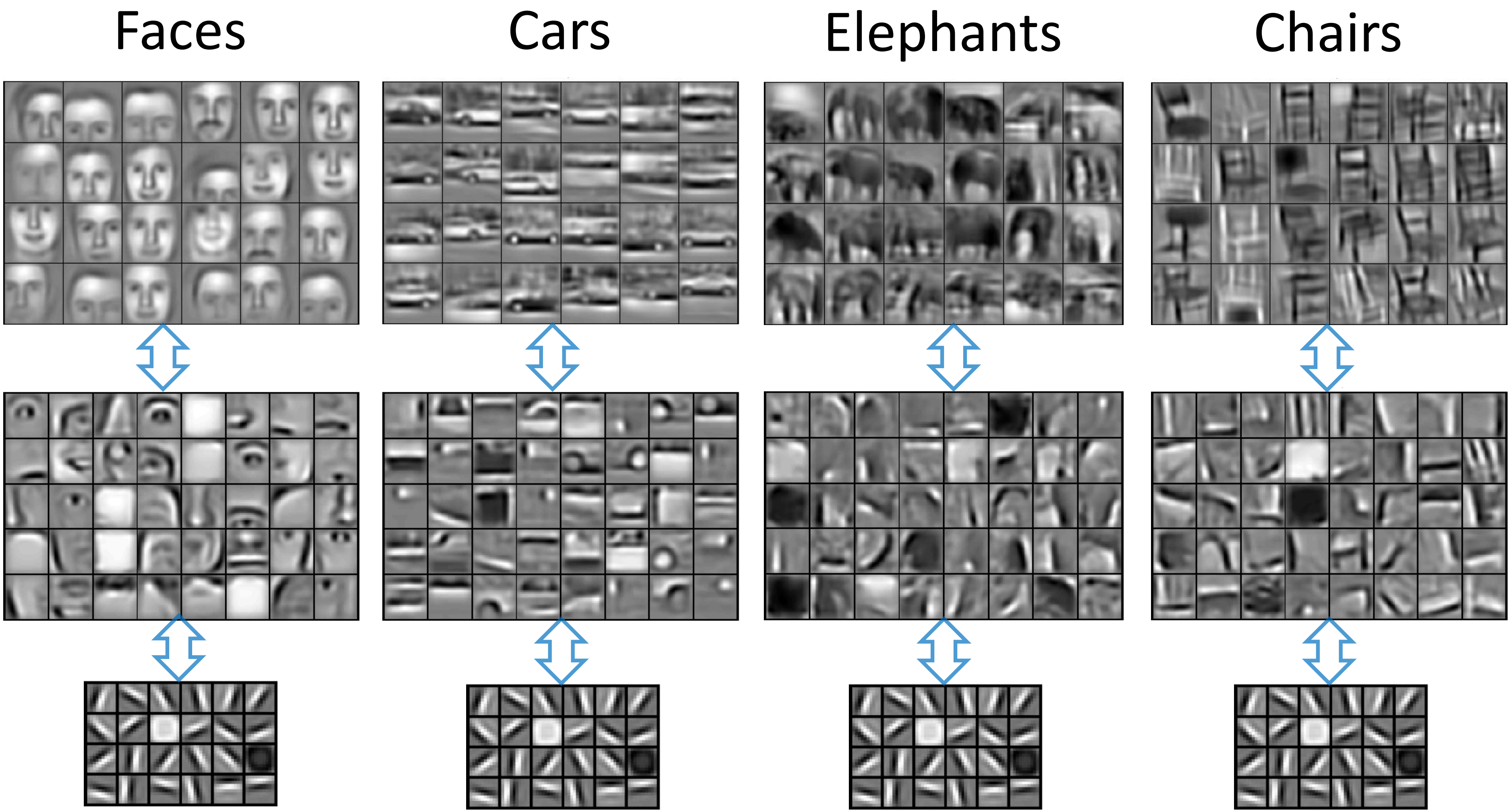
Self-taught learning for object recognition

- Caltech 101 classification: 65.4% accuracy (Convolutional DBN trained on natural images.)

Training Size	15	30
CDBN (first layer)	53.2±1.2%	60.5±1.1%
CDBN (first+second layers)	57.7±1.5%	65.4±0.5%
Raina et al. (2007)	46.6%	-
Ranzato et al. (2007)	-	54.0%
Mutch and Lowe (2006)	51.0%	56.0%
Lazebnik et al. (2006)	54.0%	64.6%
Zhang et al. (2006)	59.0±0.56%	66.2±0.5%

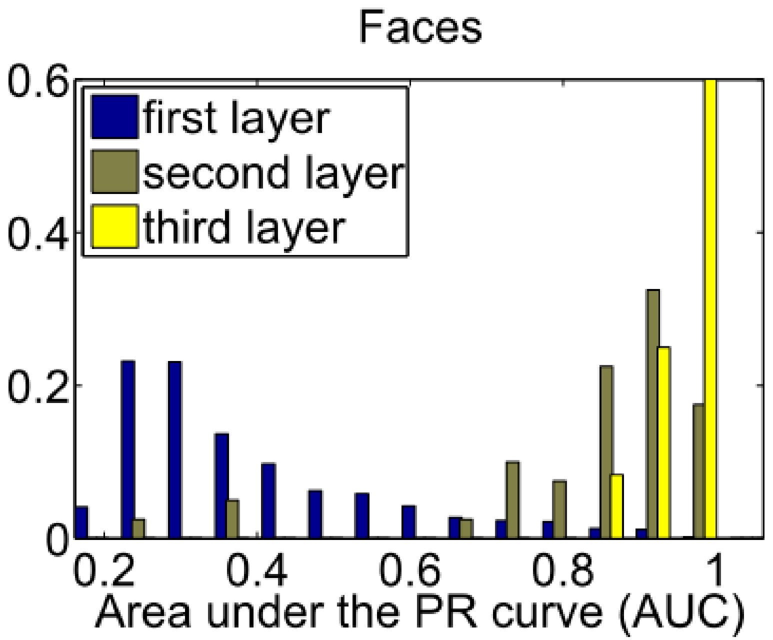
- Our model is also comparable to the results using state-of-the-art single features (e.g., SIFT).

Unsupervised learning of object-parts



Quantitative evaluation

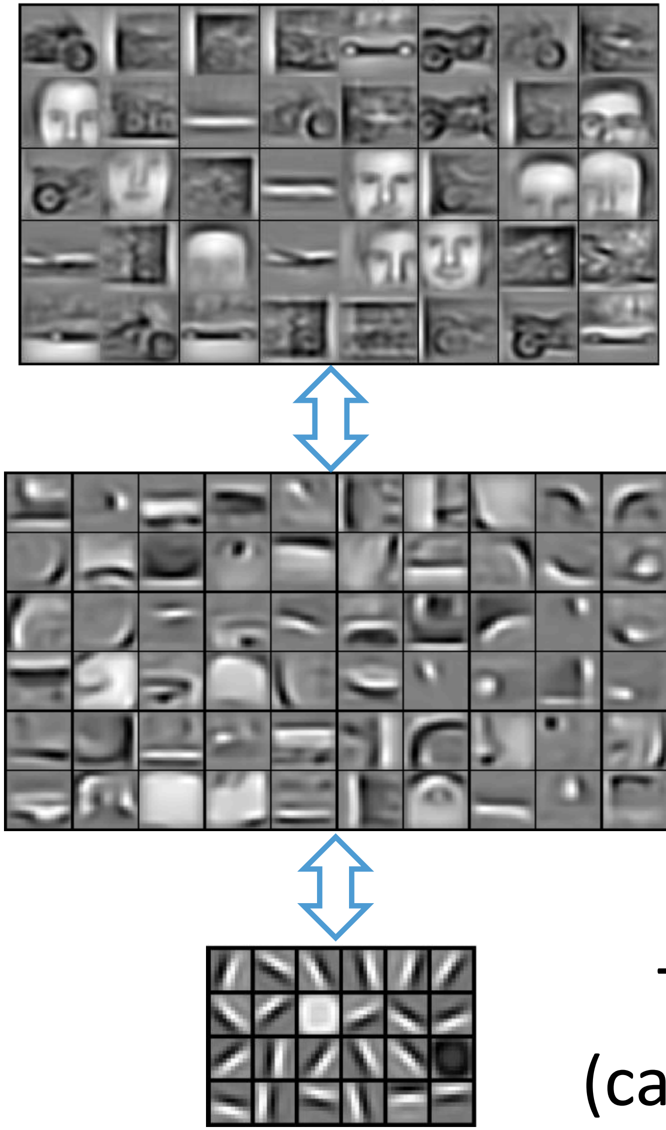
- For each feature, measure area under precision-recall curve (AUC-PR, or “average precision”) for binary classification (faces vs. non-faces).



Features	Avg. AUC-PR
First layer	0.39 ± 0.17
Second layer	0.86 ± 0.13
Third layer	$\mathbf{0.95 \pm 0.03}$

- The higher layers are informative for object class.

Unsupervised learning of object-parts



“Grouping” the object parts

(highly specific)

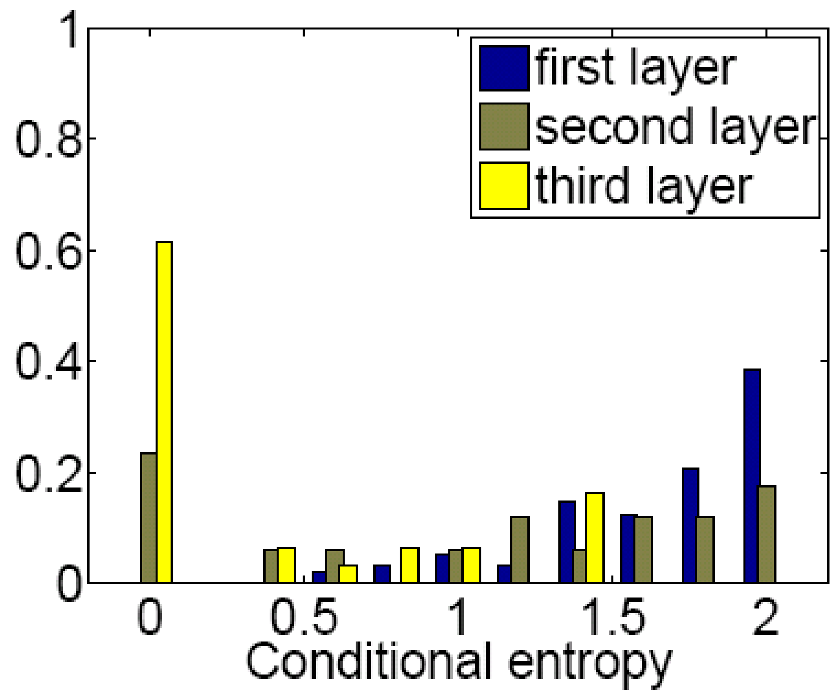
object-specific features

& shared features

Trained from multiple classes
(cars, faces, motorbikes, airplanes)

Quantitative evaluation

- Conditional entropy: H(Class|"feature active")



- The higher layers are more object specific.

Hierarchical Probabilistic Inference

- Generating posterior samples from faces by “filling in” experiments (cf. Lee and Mumford, 2003).
- Combines bottom-up and top-down inference.

Input images

Samples from feed-forward inference (control)

Samples from full posterior inference

Summary

- Convolutional Restricted Boltzmann Machine
  - Probabilistic max-pooling
- Convolutional Deep Belief Networks
  - Scalable to realistic image sizes
  - Discovers hierarchical object-part representation
  - Excellent performance in object recognition tasks
  - Hierarchical probabilistic inference by combining bottom-up and top-down information

Thank you!