Introduction to Deep Learning and its applications in Computer Vision

Xiaogang Wang

Department of Electronic Engineering,
The Chinese University of Hong Kong

Outline

- Introduction to deep learning
- Deep learning for object recognition
- Deep learning for object segmentation
- Deep learning for object detection
- Open questions and future works

Part I: Introduction to Deep Learning

- Historical review of deep learning
- Introduction to classical deep models
- Why does deep learning work?

Machine Learning

$$x \longrightarrow \boxed{F(x)} \longrightarrow y \begin{cases} \text{Class label (Classification)} \\ \text{Vector (Estimation)} \end{cases}$$

Object recognition $\longrightarrow$ {dog, cat, horse, flower, …}

Low-resolution image

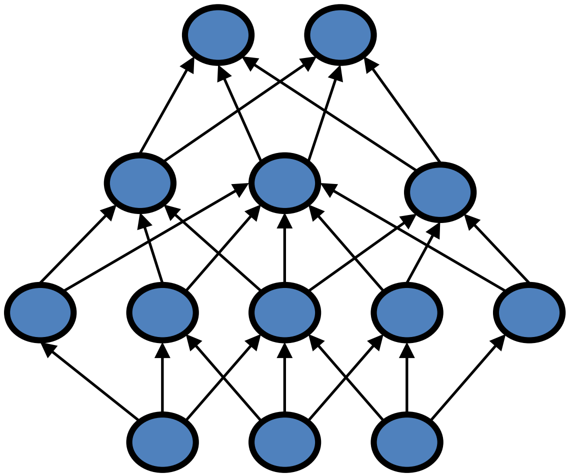
Super resolution $\longrightarrow$

High-resolution image

Neural network
Back propagation

Nature

1986

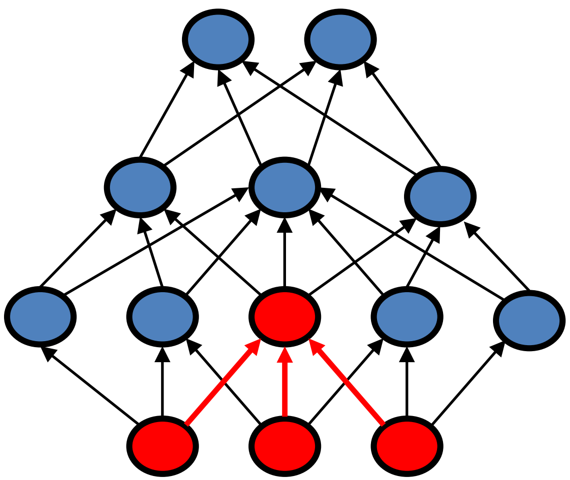


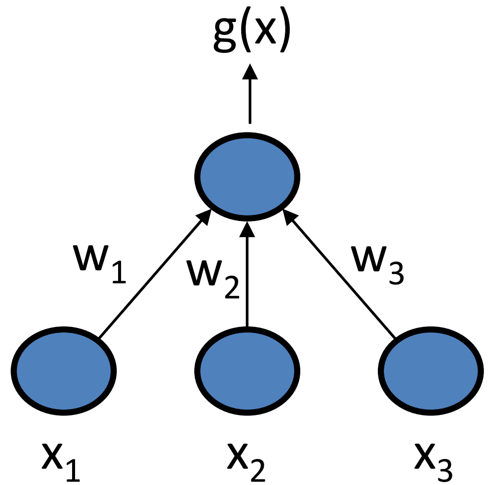
- Solve general learning problems
- Tied with biological system

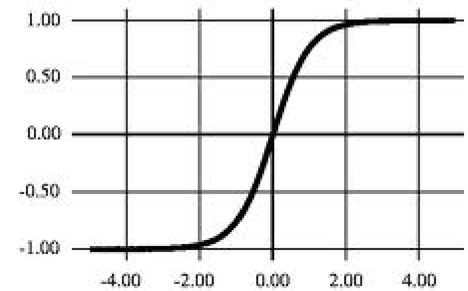
Neural network
Back propagation

Nature

1986

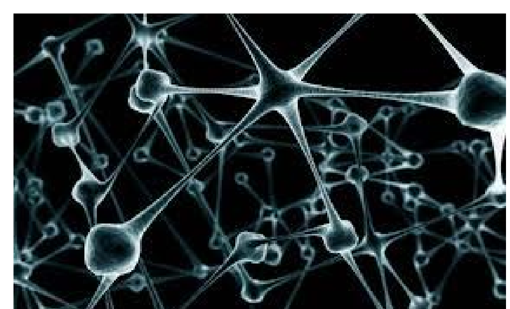






$f(net)$

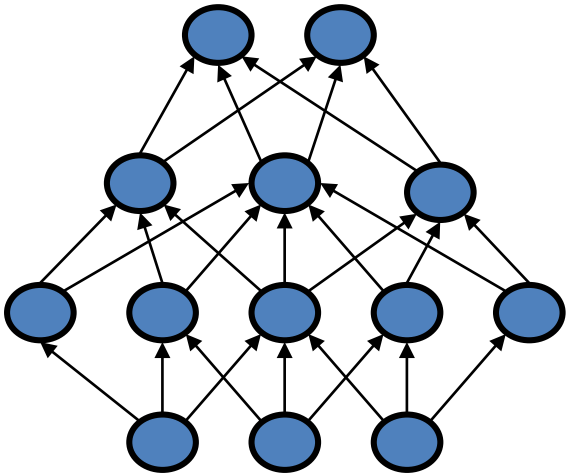
$$g(\mathbf{x}) = f(\sum_{i=1}^{d} x_i w_i + w_0) = f(\mathbf{w}^t\mathbf{x})$$



Neural network
Back propagation

Nature

1986



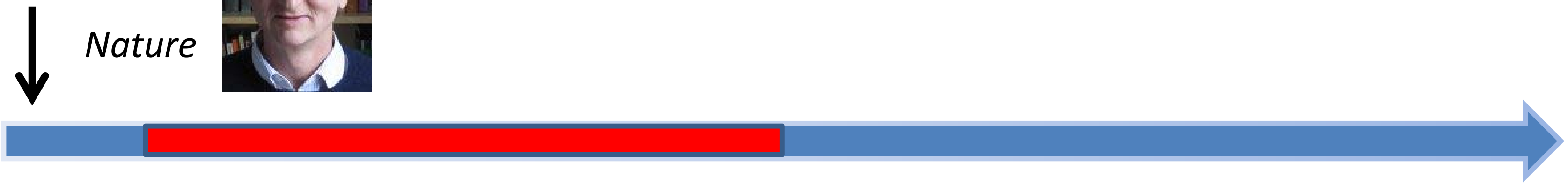
- Solve general learning problems
- Tied with biological system

But it is given up…

- Hard to train
- Insufficient computational resources
- Small training sets
- Does not work well

Neural network
Back propagation

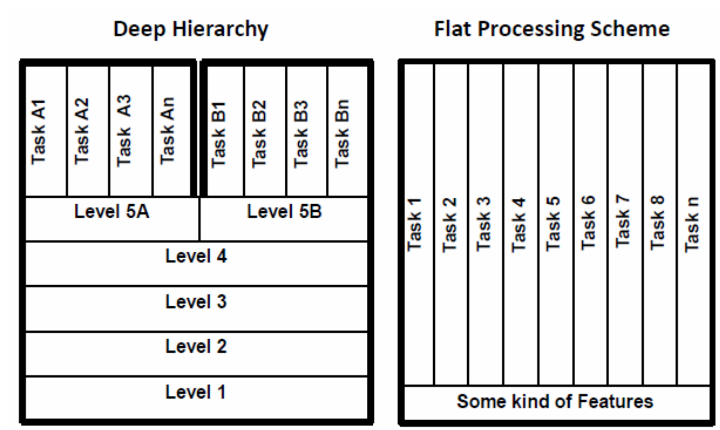
Nature



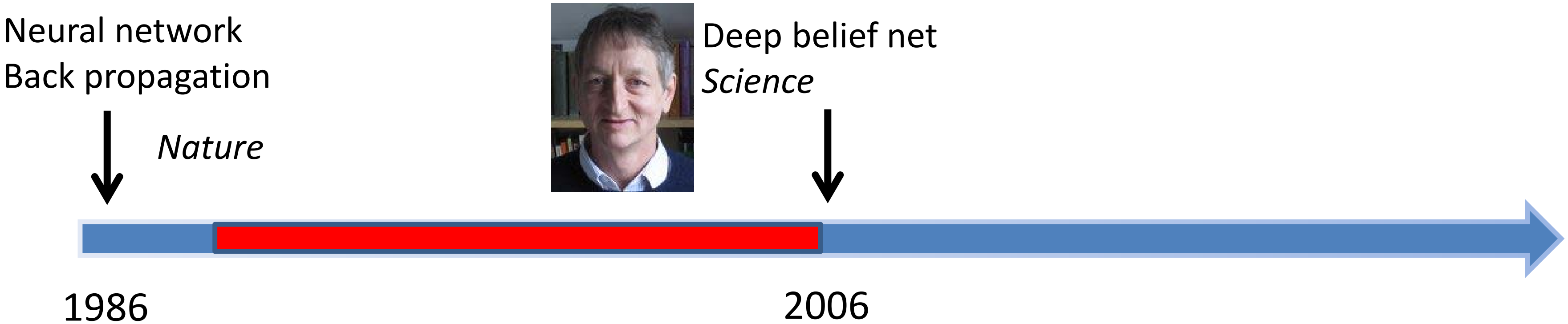
1986 2006

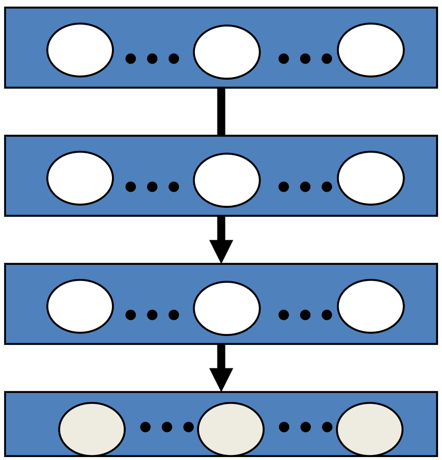
- SVM
- Boosting
- Decision tree
- KNN
- …

- Flat structures
- Loose tie with biological systems
- Specific methods for specific tasks
  - Hand crafted features (GMM-HMM, SIFT, LBP, HOG)



Kruger et al. TPAMI’13



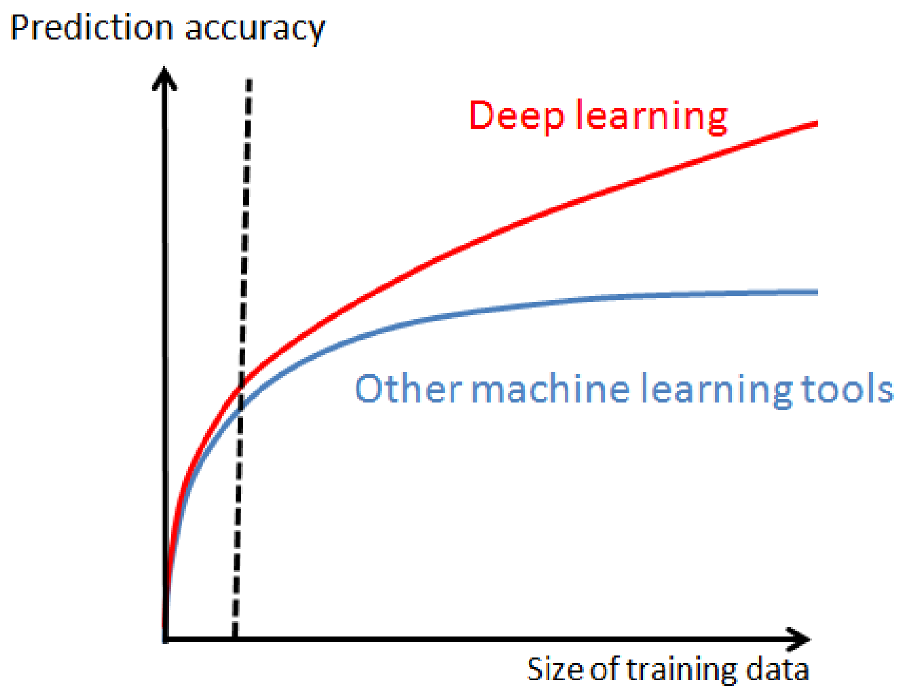


- Unsupervised & Layer-wised pre-training
- Better designs for modeling and training (normalization, nonlinearity, dropout)
- New development of computer architectures
  - GPU
  - Multi-core computer systems
- Large scale databases

Big Data !

Machine Learning with Big Data

- Machine learning with small data: overfitting, reducing model complexity (capacity)
- Machine learning with big data: underfitting, increasing model complexity, optimization, computation resource



How to increase model capacity?

Curse of dimensionality

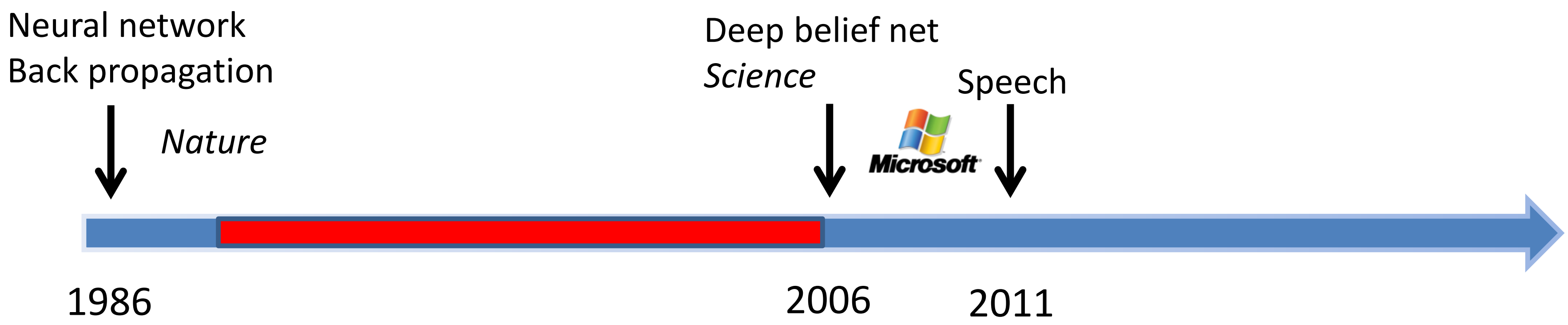
⇩

Blessing of dimensionality

⇩

Learning hierarchical feature transforms
(Learning features with deep structures)

D. Chen, X. Cao, F. Wen, and J. Sun. Blessing of dimensionality: Highdimensional feature and its efficient compression for face verification. In Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition, 2013.

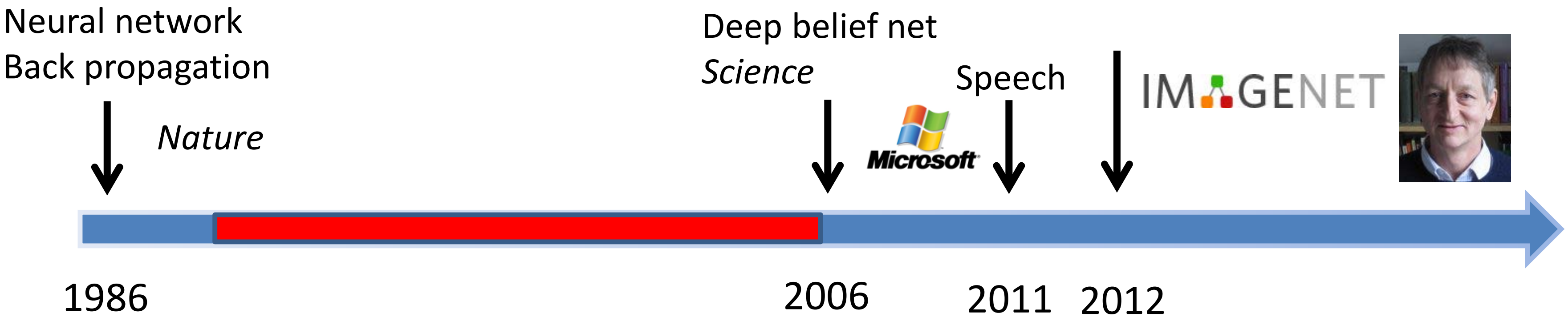


deep learning results

task	hours of training data	DNN-HMM	GMM-HMM with same data
Switchboard (test set 1)	309	18.5	27.4
Switchboard (test set 2)	309	16.1	23.6
English Broadcast News	50	17.5	18.8
Bing Voice Search (Sentence error rates)	24	30.4	36.2
Google Voice Input	5,870	12.3	
Youtube	1,400	47.6	52.3

Deep Networks Advance State of Art in Speech

Deep Learning leads to breakthrough in speech recognition at MSR.



Rank	Name	Error rate	Description
1	**U. Toronto**	0.15315	Deep learning
2	U. Tokyo	0.26172	Hand-crafted features and learning models. Bottleneck.
3	U. Oxford	0.26979	
4	Xerox/INRIA	0.27058	

Object recognition over 1,000,000 images and 1,000 categories (2 GPU)

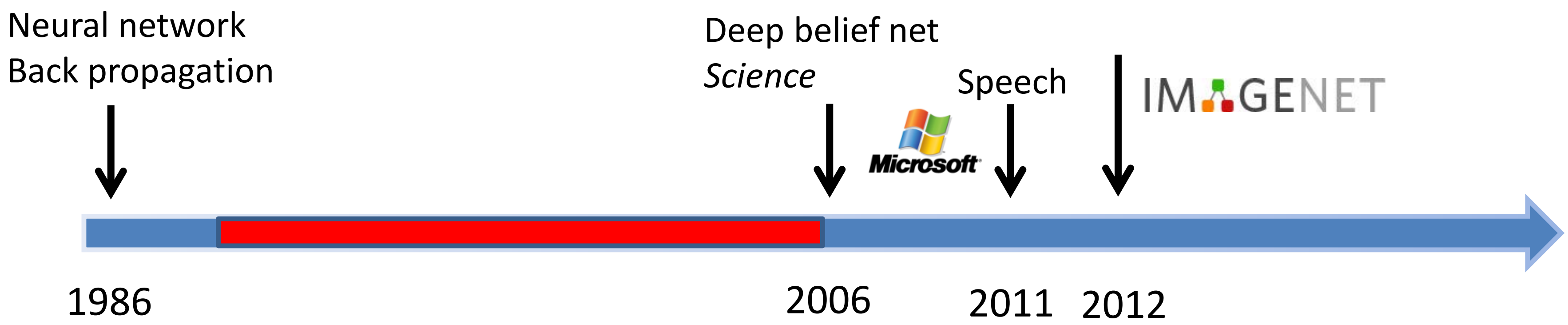
A. Krizhevsky, L. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," NIPS, 2012.

Examples from ImageNet

1000 object classes that we recognize

poster created by Fengjun Lv using VIPBase

images courtesy of ImageNet (http://www.image-net.org/challenges/LSVRC/2010/index)



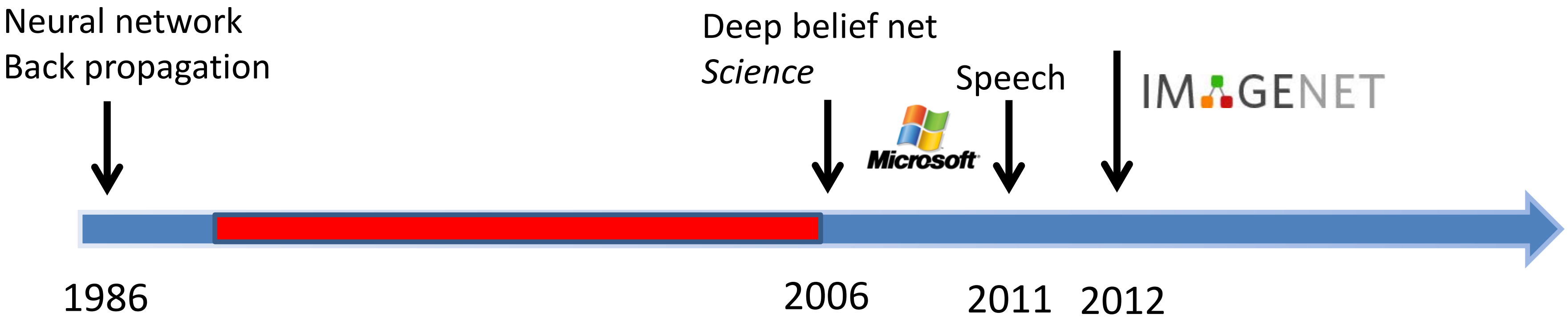
- ImageNet 2013 – image classification challenge

Rank	Name	Error rate	Description
1	NYU	0.11197	Deep learning
2	NUS	0.12535	Deep learning
3	Oxford	0.13555	Deep learning

MSRA, IBM, Adobe, NEC, Clarifai, Berkley, U. Tokyo, UCLA, UIUC, Toronto …. Top 20 groups all used deep learning

- ImageNet 2013 – object detection challenge

Rank	Name	Mean Average Precision	Description
1	UvA-Euvision	0.22581	Hand-crafted features
2	NEC-MU	0.20895	Hand-crafted features
3	NYU	0.19400	Deep learning

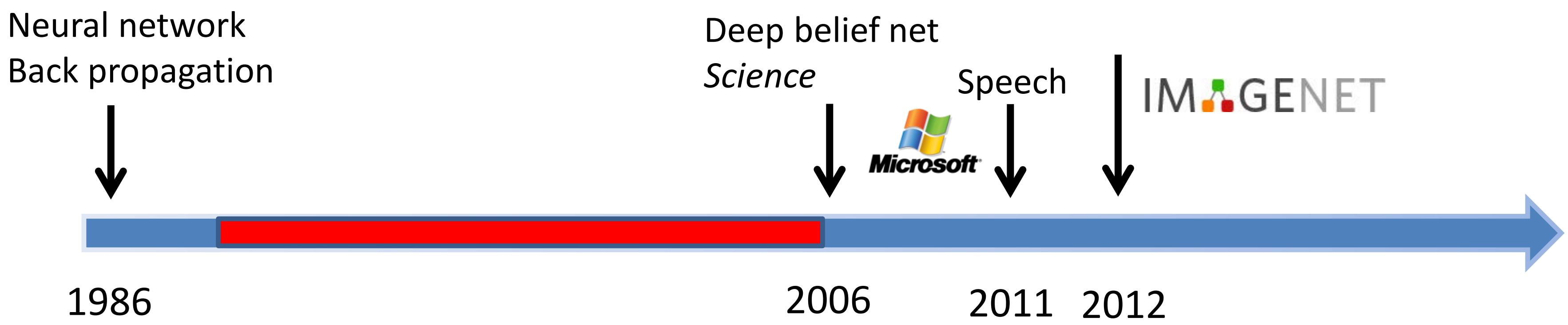


- ImageNet 2014 – Image classification challenge

Rank	Name	Error rate	Description
1	Google	0.06656	Deep learning
2	Oxford	0.07325	Deep learning
3	MSRA	0.08062	Deep learning

- ImageNet 2014 – object detection challenge

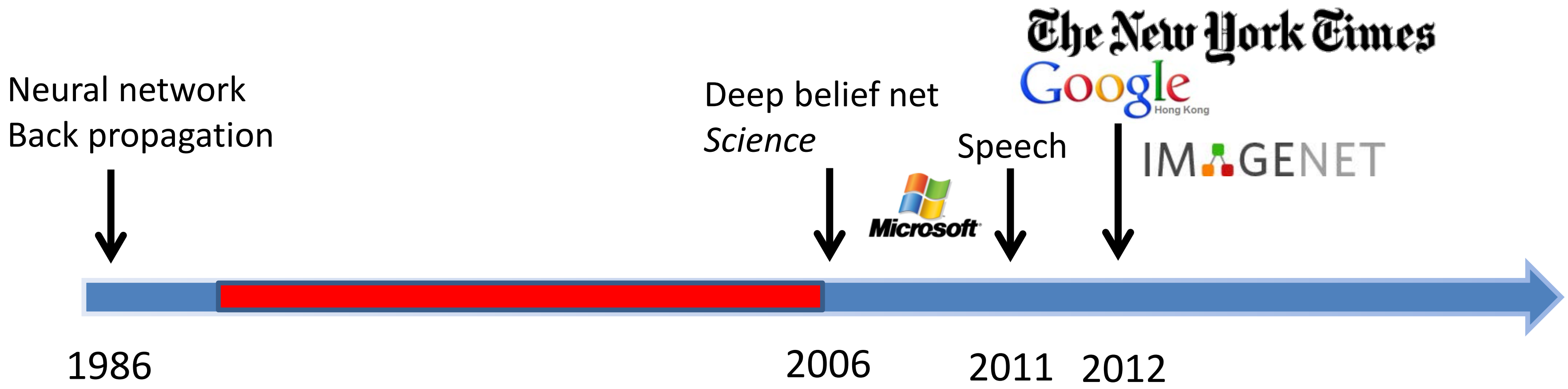
Rank	Name	Mean Average Precision	Description
1	Google	0.43933	Deep learning
2	CUHK	0.40656	Deep learning
3	DeepInsight	0.40452	Deep learning
4	UvA-Euvision	0.35421	Deep learning
5	Berkley Vision	0.34521	Deep learning



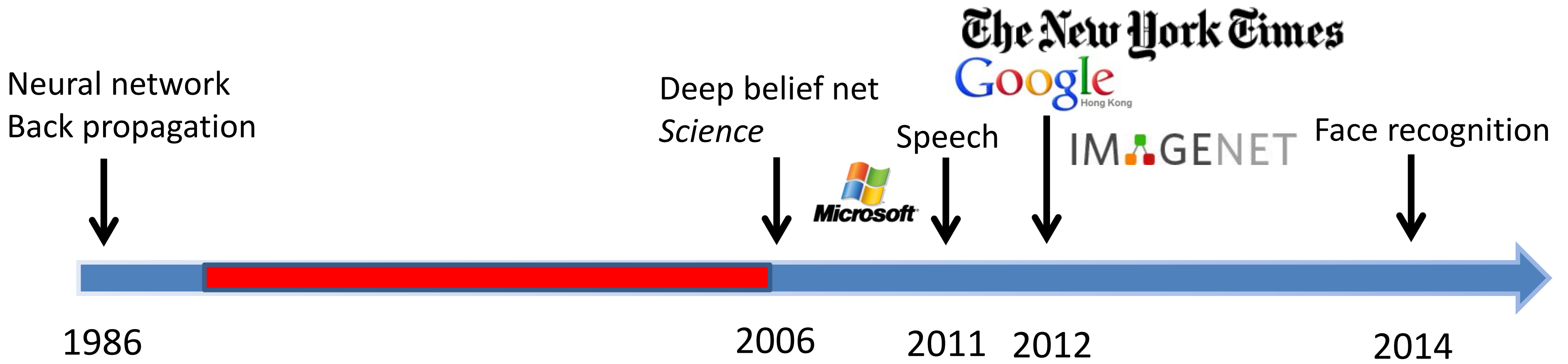
- ImageNet 2014 – object detection challenge

	GoogLeNet (Google)	DeepID-Net (CUHK)	DeepInsight	UvA-Euvision	Berkley Vision	RCNN
Model average	0.439	**0.439**	0.405	n/a	n/a	n/a
Single model	0.380	**0.427**	0.402	0.354	0.345	0.314

W. Ouyang and X. Wang et al. "DeepID-Net: deformable deep convolutional neural networks for object detection", CVPR, 2014



- Google and Baidu announced their deep learning based visual search engines (2013)
  - [Google](#)
    - "on our test set we saw **double the average precision** when compared to other approaches we had tried. We acquired the rights to the technology and went full speed ahead adapting it to run at large scale on Google's computers. We took cutting edge research straight out of an academic research lab and launched it, in just a little over six months."
  - [Baidu](#)

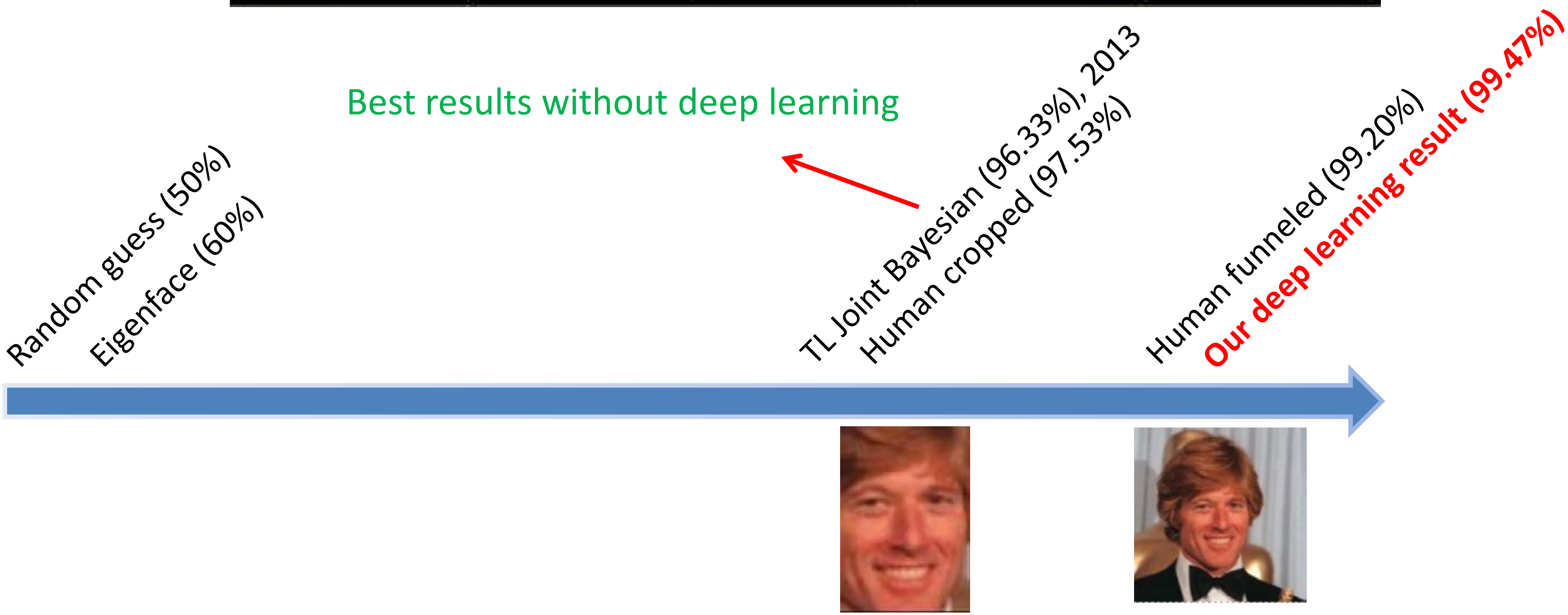


- Deep learning achieves 99.47% face verification accuracy on Labeled Faces in the Wild (LFW), higher than human performance

Y. Sun, X. Wang, and X. Tang. Deep Learning Face Representation by Joint Identification-Verification. NIPS, 2014.

Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. CVPR, 2015.

Labeled Faces in the Wild (2007)



Unrestricted, Labeled Outside Data Results

Attribute classifiers[11]	0.8525 ± 0.0060
Simile classifiers[11]	0.8414 ± 0.0041
Attribute and Simile classifiers[11]	0.8554 ± 0.0035
Multiple LE + comp[14]	0.8445 ± 0.0046
Associate-Predict[18]	0.9057 ± 0.0056
Tom-vs-Pete[23]	0.9310 ± 0.0135
Tom-vs-Pete + Attribute[23]	0.9330 ± 0.0128
combined Joint Bayesian[26]	0.9242 ± 0.0108
high-dim LBP[27]	0.9517 ± 0.0113
DFD[33]	0.8402 ± 0.0044
TL Joint Bayesian[34]	0.9633 ± 0.0108
face.com r2011b[19]	0.9130 ± 0.0030
Face++[40]	0.9727 ± 0.0065
DeepFace-ensemble[41]	0.9735 ± 0.0025
ConvNet-RBM[42]	0.9252 ± 0.0038
POOF-gradhist[44]	0.9313 ± 0.0040
POOF-HOG[44]	0.9280 ± 0.0047
FR+FCN[45]	0.9645 ± 0.0025
DeepID[46]	0.9745 ± 0.0026
GaussianFace[47]	0.9852 ± 0.0066
DeepID2[48]	0.9915 ± 0.0013

Table 6: Mean classification accuracy û and standard error of the mean S_E.

MIT Technology Review

10 BREAKTHROUGH TECHNOLOGIES 2013

Deep Learning

With massive amounts of computational power, machines can now recognize objects and translate speech in real time. Artificial intelligence is finally getting smart.

Temporary Social Media

Messages that quickly self-destruct could enhance the privacy of online communications and make people freer to be spontaneous.

Prenatal DNA Sequencing

Reading the DNA of fetuses will be the next frontier of the genomic revolution. But do you really want to know about the genetic problems or musical aptitude of your unborn child?

Additive Manufacturing

Skeptical about 3-D printing? GE, the world's largest manufacturer, is on the verge of using the technology to make jet parts.

Baxter: The Blue-Collar Robot

Rodney Brooks's newest creation is easy to interact with, but the complex innovations behind the robot show just how hard it is to get along with people.

Memory Implants

A maverick neuroscientist believes he has deciphered the code by which the brain forms long-term memories. Next: testing a prosthetic implant for people suffering from long-term memory loss

Smart Watches

The designers of the Pebble watch realized that a mobile phone is more useful if you don't have to take it out of your pocket

Ultra-Efficient Solar Power

Doubling the efficiency of a solar cell would completely change the economics of renewable energy. Nanotechnology just might make it possible

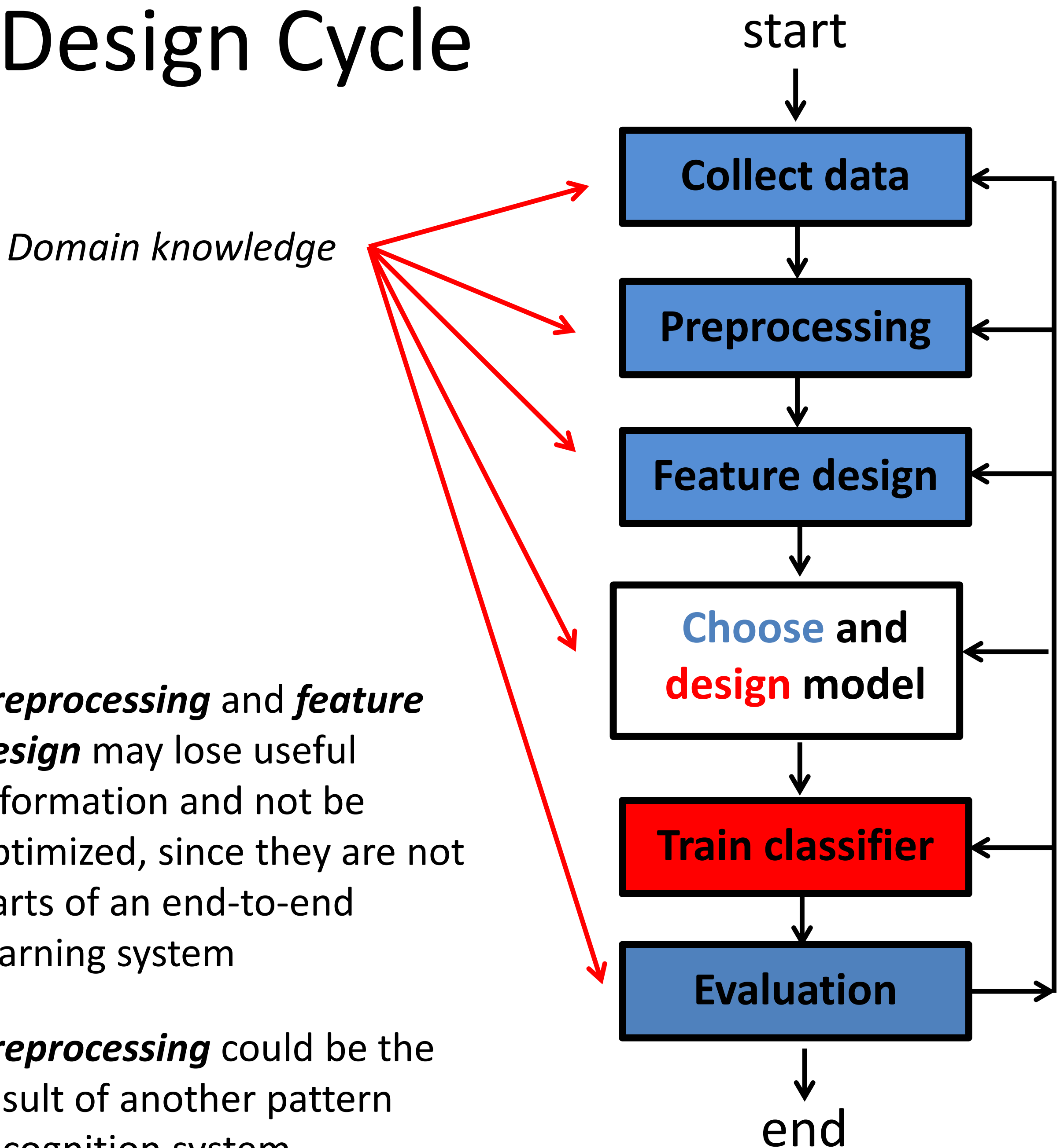
Big Data from Cheap Phones

Collecting and analyzing information from simple cell phones can provide surprising insights into how people move about and behave – and even help us understand the spread of diseases

Supergrids

A new high-power circuit breaker could finally make highly efficient DC power grids practical

Design Cycle



Interest of people working on **computer vision, speech recognition, medical image processing,…**

Interest of people working on **machine learning**

Interest of people working on **machine learning** and **computer vision, speech recognition, medical image processing,…**

Preprocessing and ***feature design*** may lose useful information and not be optimized, since they are not parts of an end-to-end learning system

Preprocessing could be the result of another pattern recognition system

(a)

(b)

Person re-identification pipeline

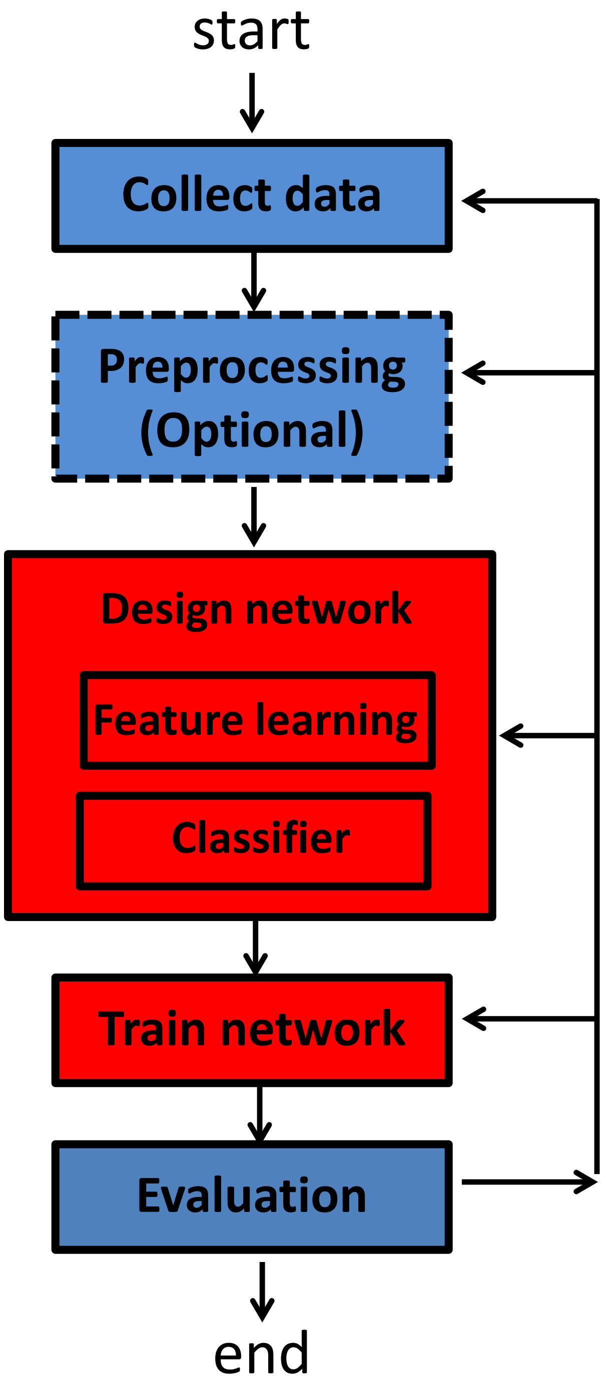
Pedestrian detection → Pose estimation → Body parts segmentation → Photometric & geometric transform → Feature extraction → Classification

Face recognition pipeline

Face alignment → Geometric rectification → Photometric rectification → Feature extraction → Classification

Design Cycle with Deep Learning

- Learning plays a bigger role in the design circle
- Feature learning becomes part of the end-to-end learning system
- Preprocessing becomes optional means that several pattern recognition steps can be merged into one end-to-end learning system
- Feature learning makes the key difference
- We underestimated the importance of data collection and evaluation



What makes deep learning successful in computer vision?

Li Fei-Fei

IMAGENET

Geoffrey Hinton

Data collection

One million images with labels

Evaluation task

Predict 1,000 image categories

Deep learning

CNN is not new

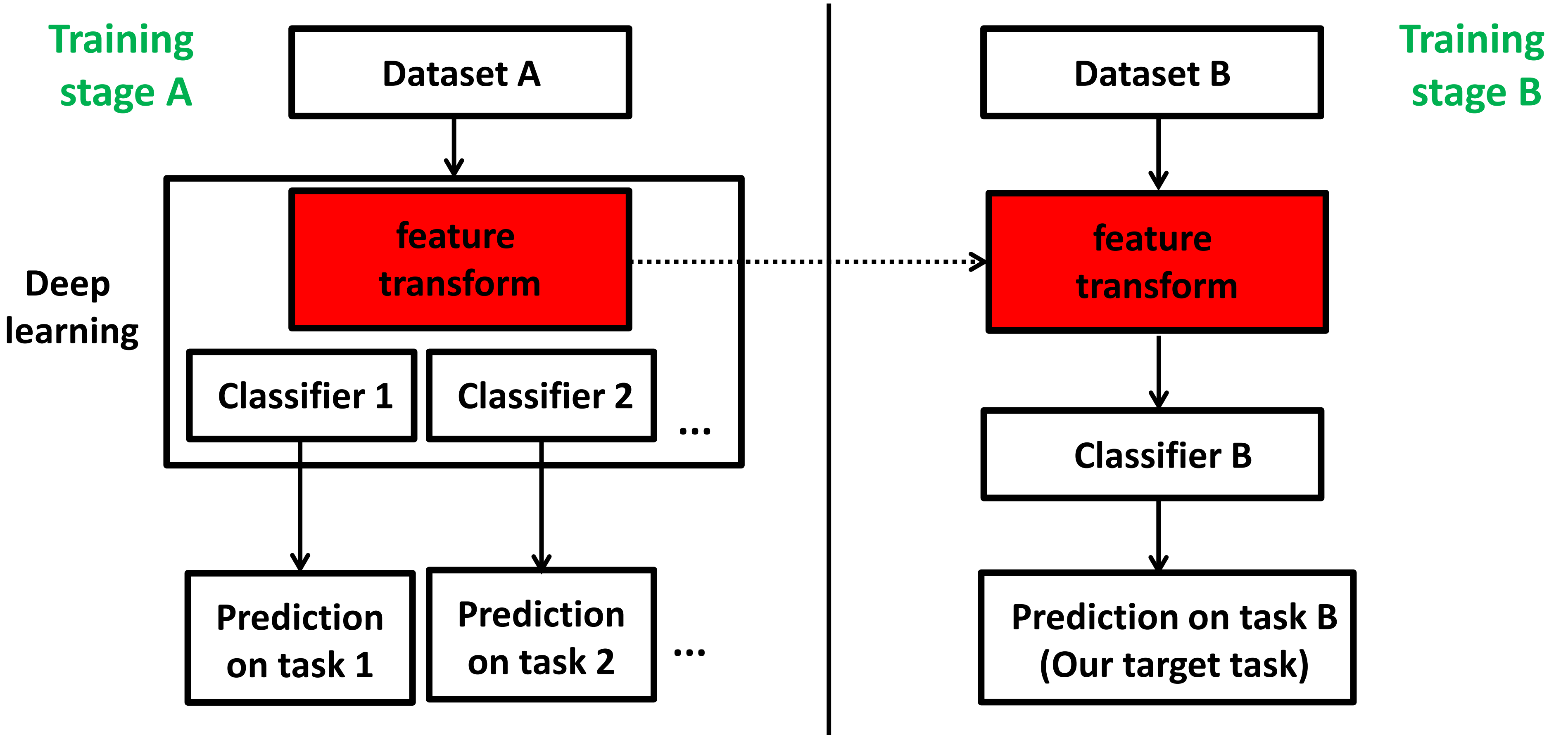
Design network structure

New training strategies

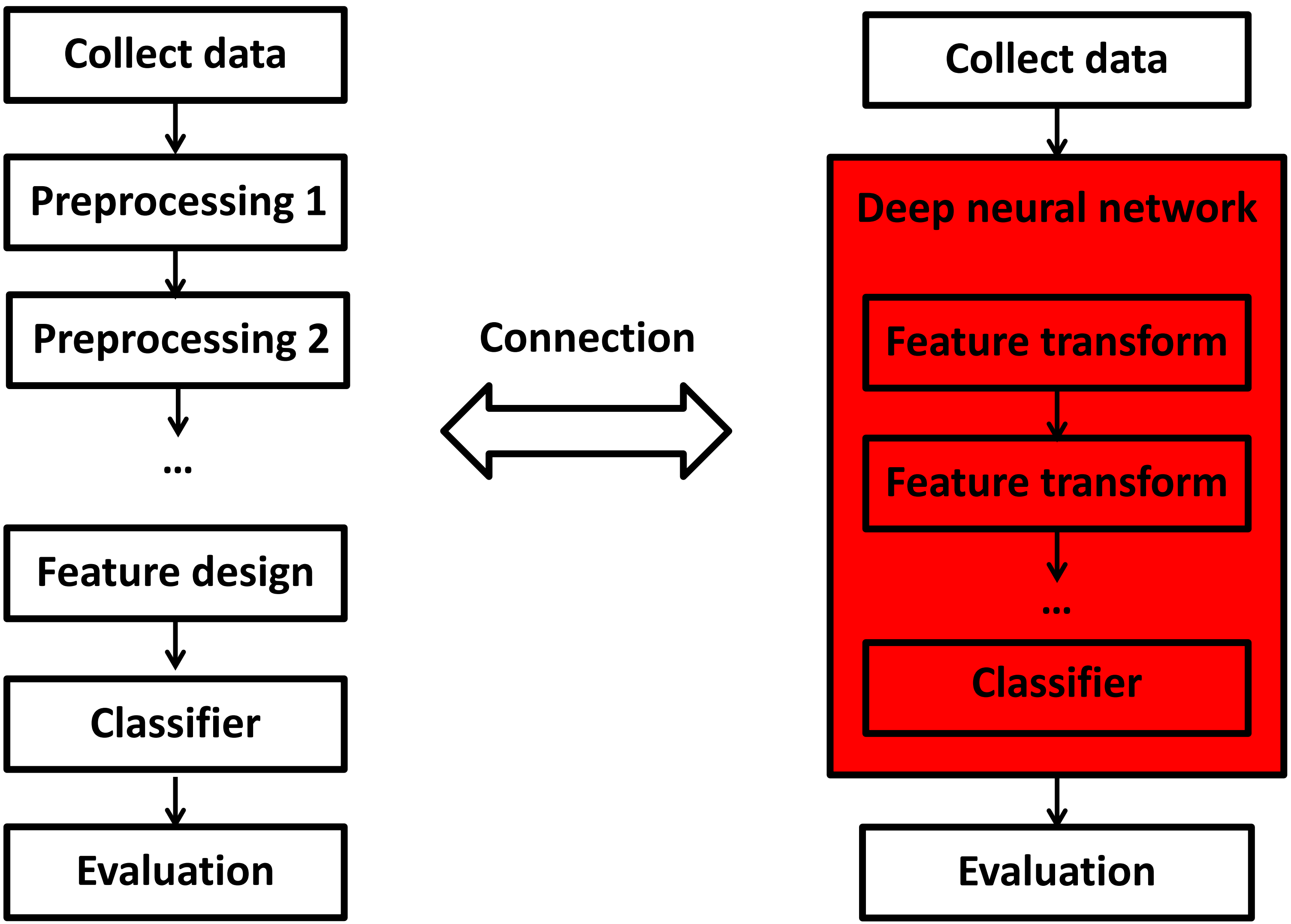
Feature learned from ImageNet can be well generalized to other tasks and datasets!

Learning features and classifiers separately

- Not all the datasets and prediction tasks are suitable for learning features with deep models



Deep learning can be treated as a language to described the world with great flexibility



Introduction to Deep Learning

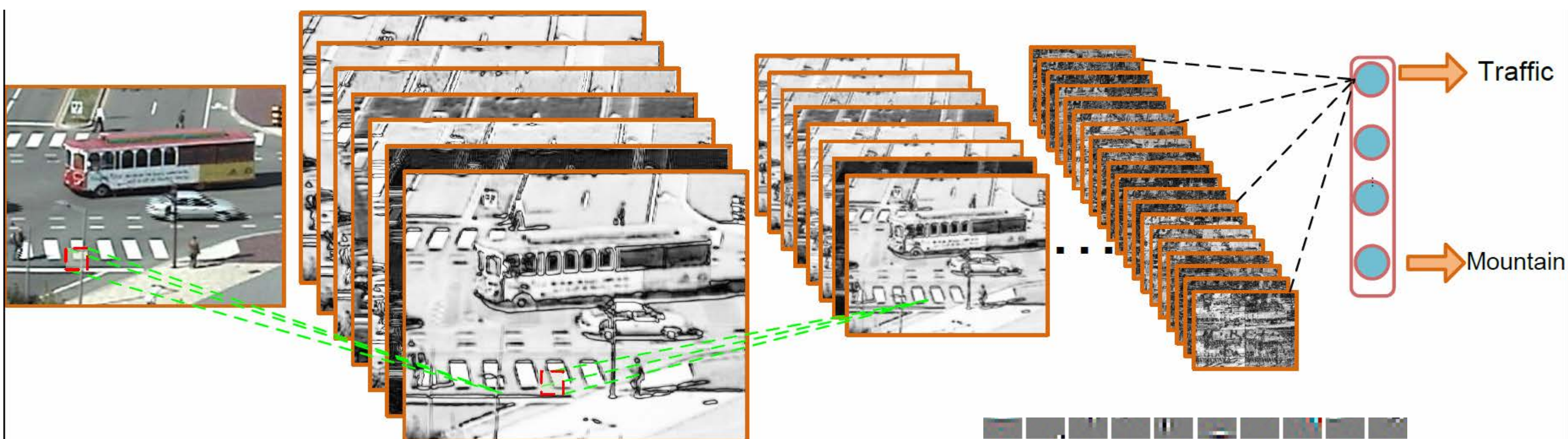
- Historical review of deep learning
- **Introduction to classical deep models**
- Why does deep learning work?

Introduction on Classical Deep Models

- Convolutional Neural Networks (CNN)
  - Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based Learning Applied to Document Recognition,” Proceedings of the IEEE, Vol. 86, pp. 2278-2324, 1998.
- Deep Belief Net (DBN)
  - G. E. Hinton, S. Osindero, and Y. Teh, “A Fast Learning Algorithm for Deep Belief Nets,” Neural Computation, Vol. 18, pp. 1527-1544, 2006.
- Auto-encoder
  - G. E. Hinton and R. R. Salakhutdinov, “Reducing the Dimensionality of Data with Neural Networks,” Science, Vol. 313, pp. 504-507, July 2006.

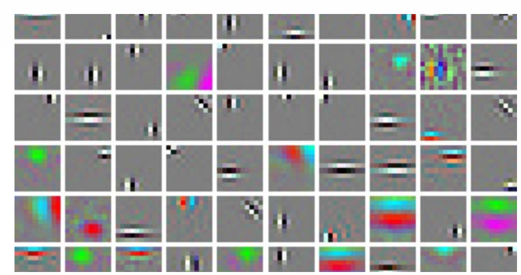
Classical Deep Models

- Convolutional Neural Networks (CNN)
  - First proposed by Fukushima in 1980
  - Improved by LeCun, Bottou, Bengio and Haffner in 1998



Convolution

Pooling

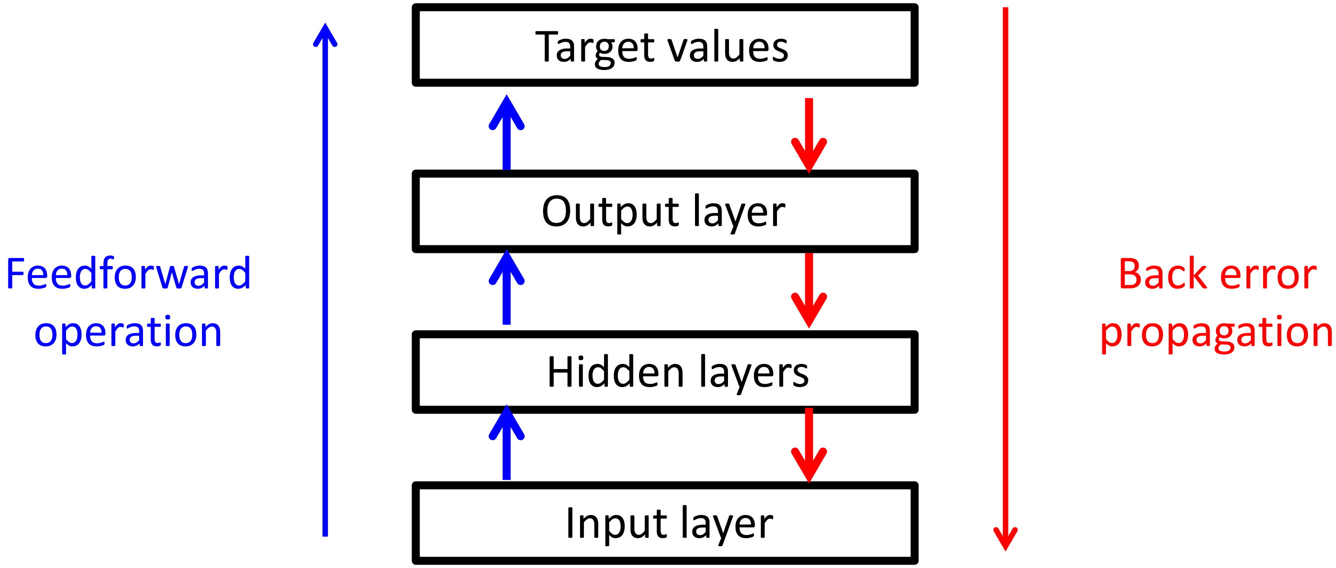


Learned filters

Backpropagation

$$\mathbf{W} \leftarrow \mathbf{W} - \eta \bigtriangledown J(\mathbf{W})$$

$\mathbf{W}$ is the parameter of the network; J is the objective function

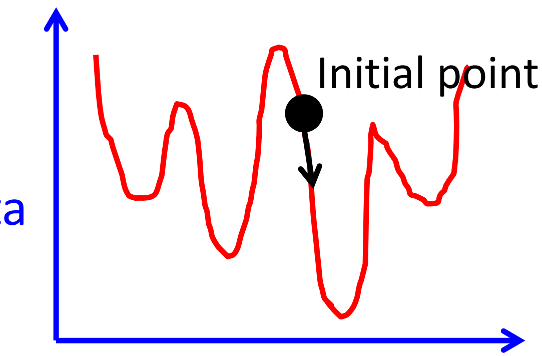


D. E. Rumelhart, G. E. Hinton, R. J. Williams, “Learning Representations by Back-propagation Errors,” Nature, Vol. 323, pp. 533-536, 1986.

Classical Deep Models

- Deep belief net
  - Hinton'06

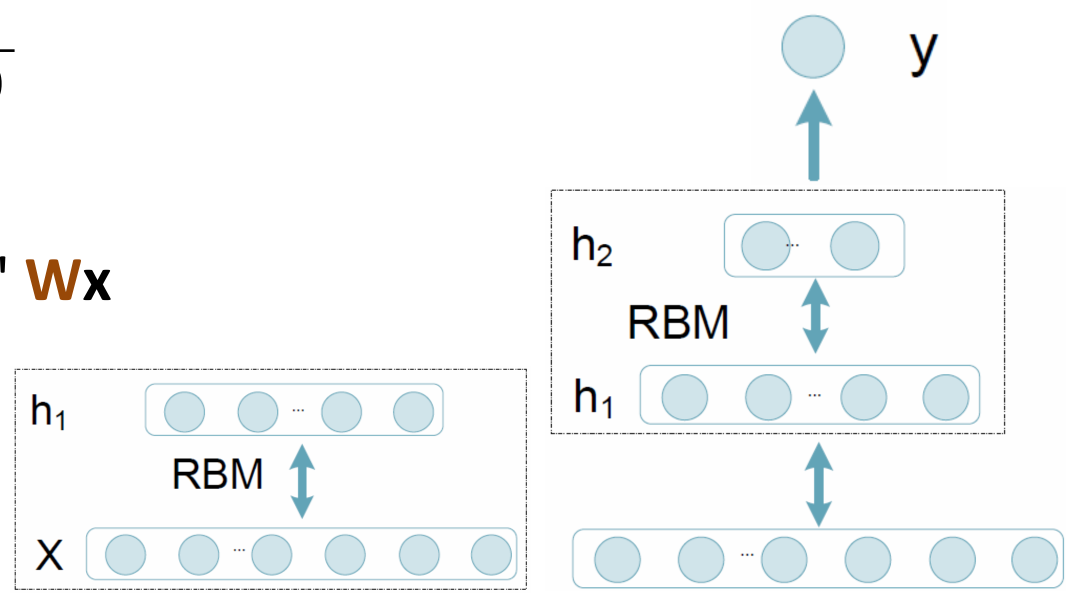
Pre-training:
- Good initialization point
- Make use of unlabeled data



$$P(\mathbf{x},\mathbf{h}_1,\mathbf{h}_2) = p(\mathbf{x}|\mathbf{h}_1)\, p(\mathbf{h}_1,\mathbf{h}_2)$$

$$P(\mathbf{x},\mathbf{h}_1) = \frac{e^{-E(\mathbf{x},\mathbf{h}_1)}}{\sum_{\mathbf{x},\mathbf{h}_1} e^{-E(\mathbf{x},\mathbf{h}_1)}}$$

$$E(\mathbf{x},\mathbf{h}_1) = \mathbf{b}'\,\mathbf{x} + \mathbf{c}'\,\mathbf{h}_1 + \mathbf{h}_1'\,\mathbf{W}\mathbf{x}$$



Classical Deep Models

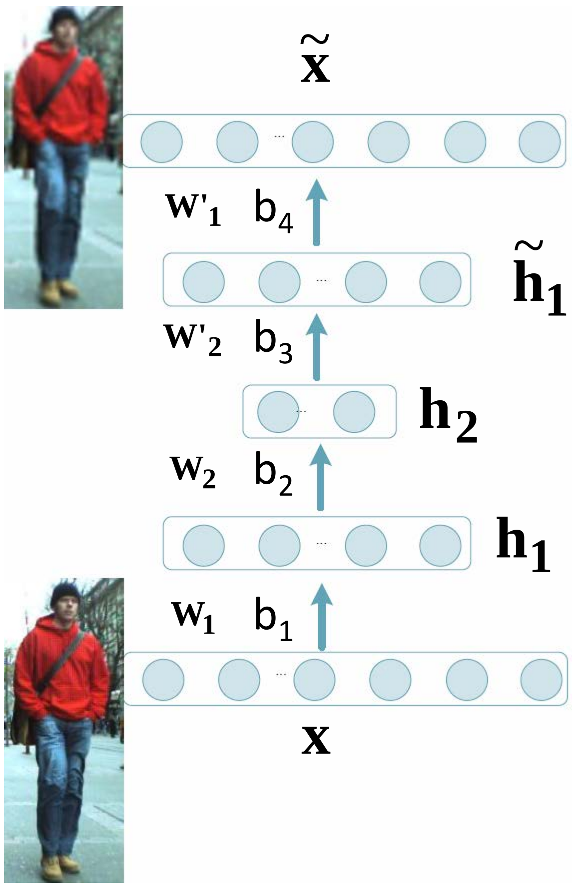
- Auto-encoder
  - Hinton and Salakhutdinov 2006

Encoding: $\mathbf{h}_1 = \sigma(\mathbf{W}_1\mathbf{x}+b_1)$

$\mathbf{h}_2 = \sigma(\mathbf{W}_2\mathbf{h}_1+b_2)$

Decoding: $\tilde{\mathbf{h}}_1 = \sigma(\mathbf{W}'_2\mathbf{h}_2+b_3)$

$\tilde{\mathbf{x}} = \sigma(\mathbf{W}'_1\mathbf{h}_1+b_4)$



Introduction to Deep Learning

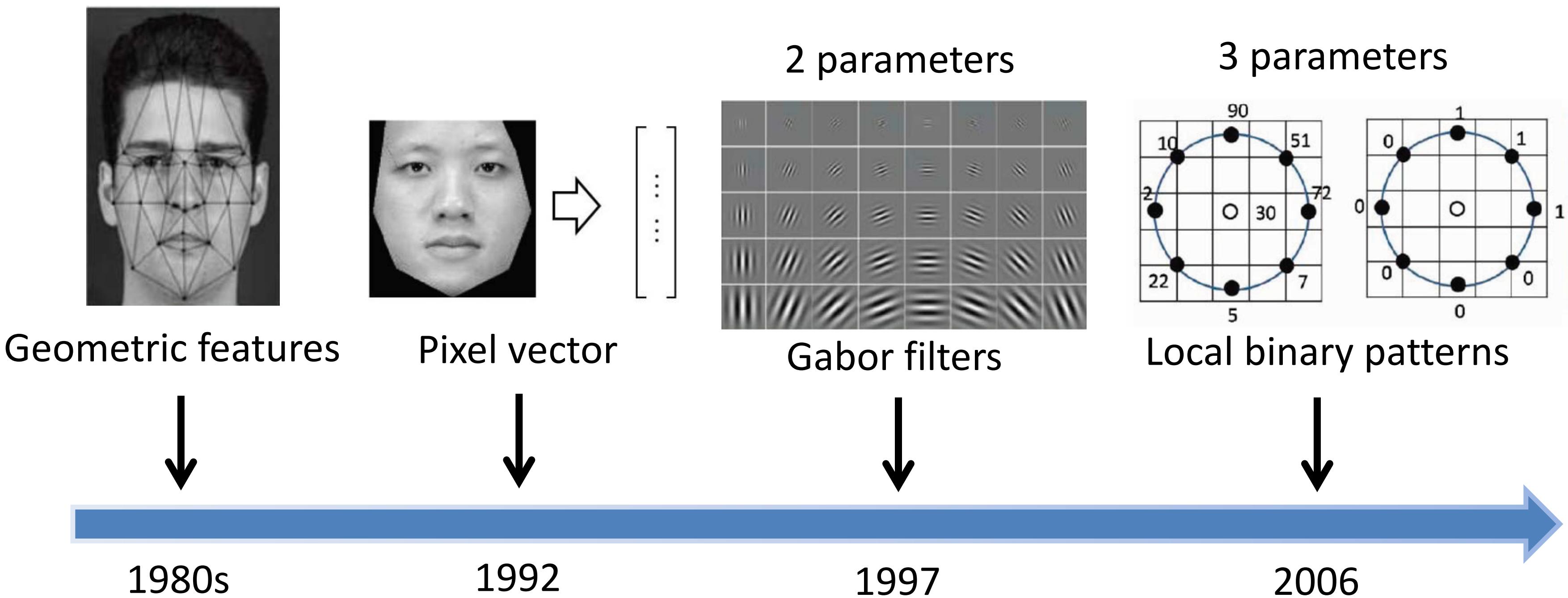
- Historical review of deep learning
- Introduction to classical deep models
- **Why does deep learning work?**

Feature Learning vs Feature Engineering

Feature Engineering

- The performance of a pattern recognition system heavily depends on feature representations
- Manually designed features dominate the applications of image and video understanding in the past
  - Reply on human domain knowledge much more than data
  - Feature design is separate from training the classifier
  - If handcrafted features have multiple parameters, it is hard to manually tune them
  - Developing effective features for new applications is slow

Handcrafted Features for Face Recognition



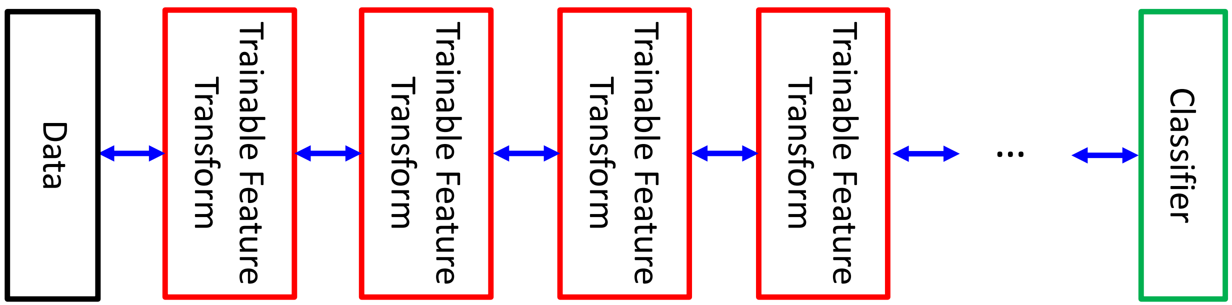
Feature Learning

- Learning transformations of the data that make it easier to extract useful information when building classifiers or predictors
  - Jointly learning feature transformations and classifiers makes their integration optimal
  - Learn the values of a huge number of parameters in feature representations
  - Faster to get feature representations for new applications
  - Make better use of big data

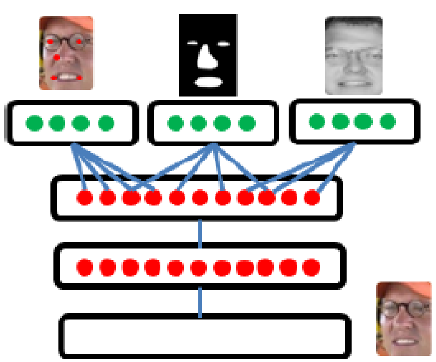
Deep Learning Means Feature Learning

- Deep learning is about learning hierarchical feature representations

$$\mathbf{y} = F(\mathbf{W}^k \cdot F(\mathbf{W}^{k-1} \cdot F(\ldots F(\mathbf{W}^0 \cdot \mathbf{x}))$$



- Good feature representations should be able to disentangle multiple factors coupled in the data

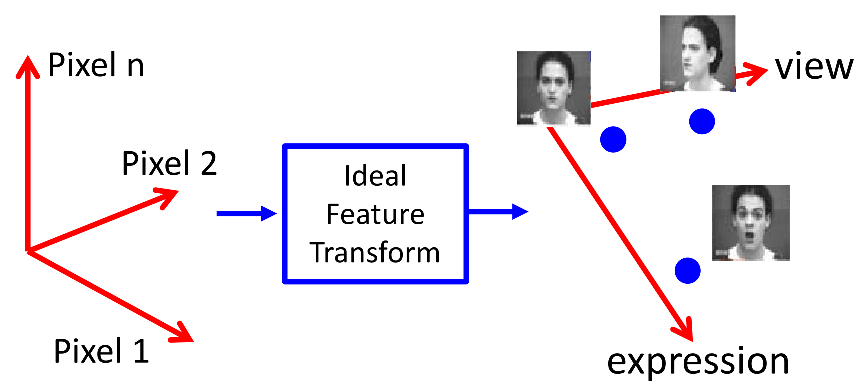


Identity: face recognition

Pose: pose estimation

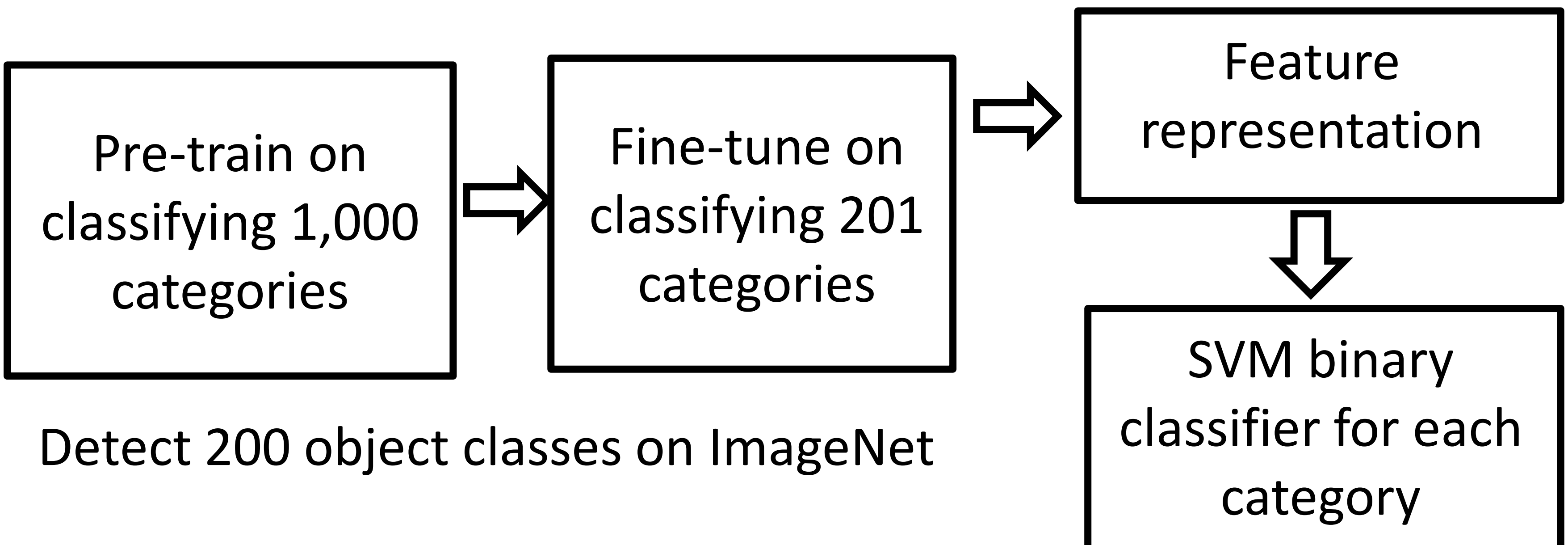
Expression: expression recognition

Age: age estimation



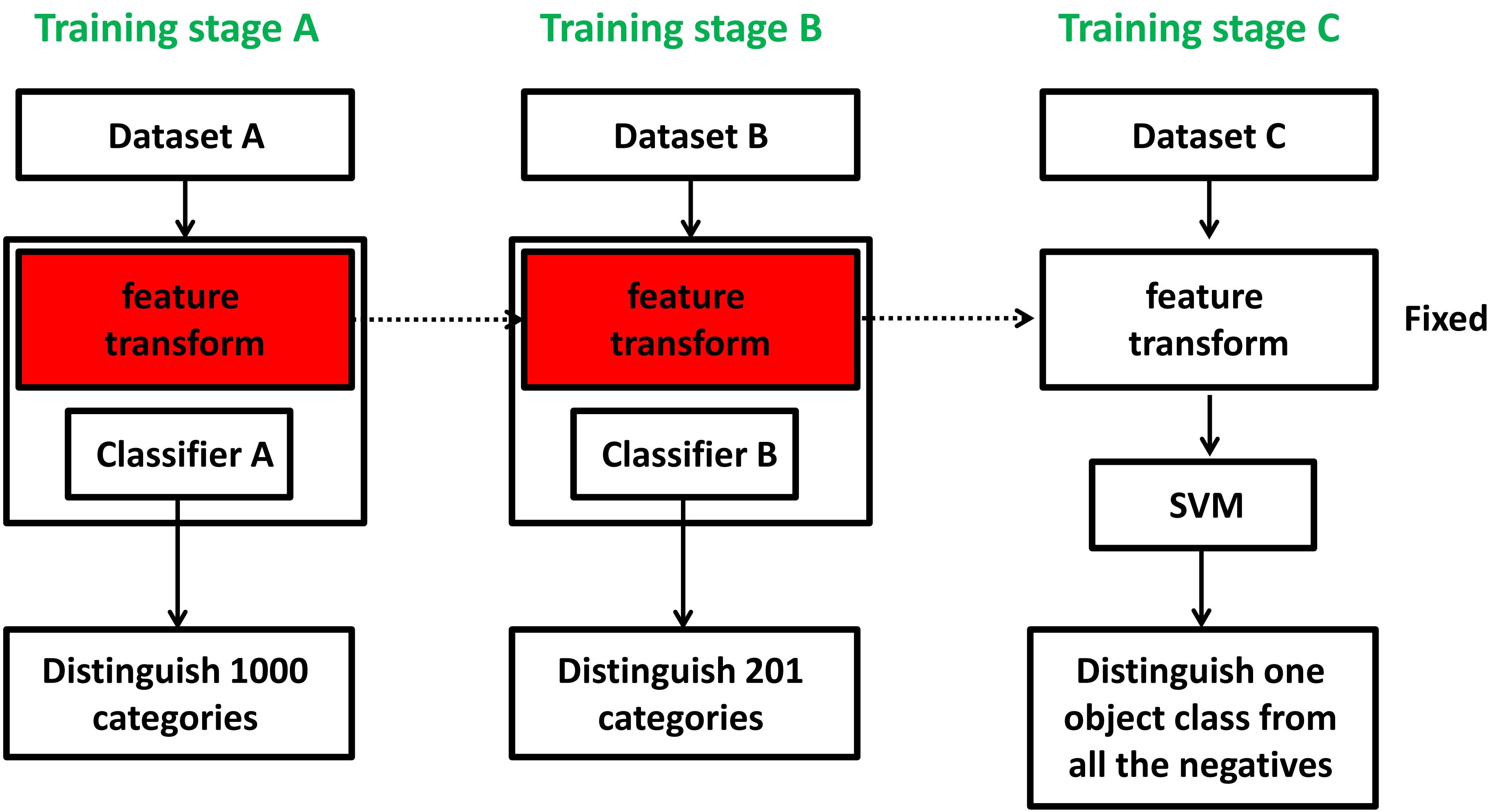
Deep Learning Means Feature Learning

- How to effectively learn features with deep models
  - With challenging tasks
  - Predict high-dimensional vectors



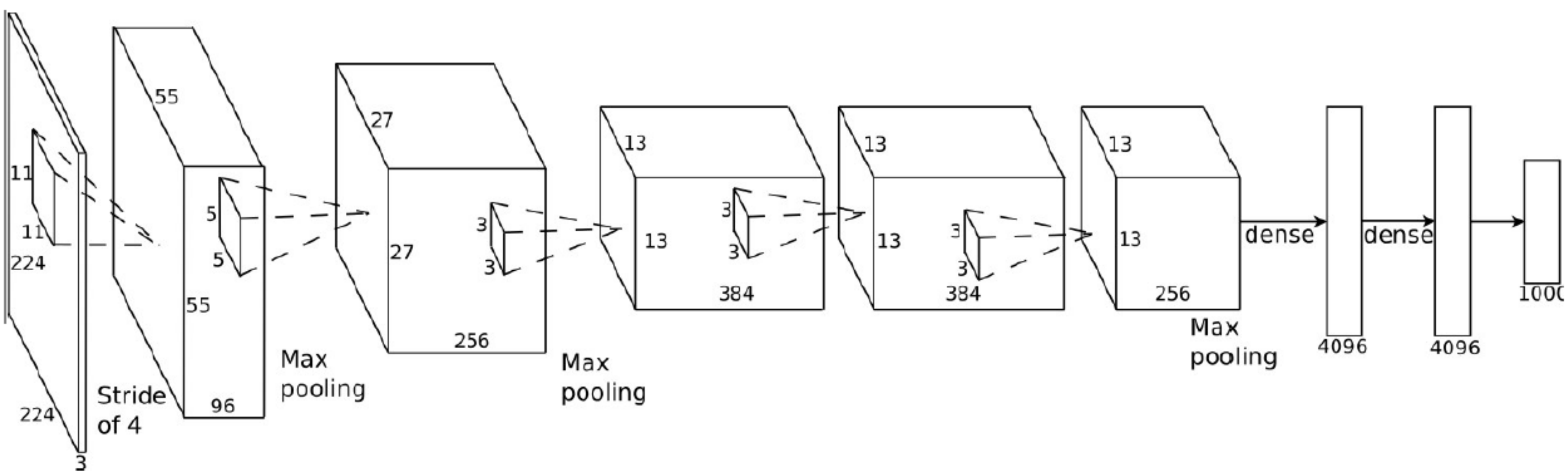
Detect 200 object classes on ImageNet

W. Ouyang and X. Wang et al. "DeepID-Net: deformable deep convolutional neural networks for object detection", CVPR, 2015


Training stage A
Training stage B
Training stage C
Dataset A
Dataset B
Dataset C
feature transform
feature transform
feature transform
Fixed
Classifier A
Classifier B
SVM
Distinguish 1000 categories
Distinguish 201 categories
Distinguish one object class from all the negatives

Example 1: deep learning generic image features

- Hinton group's groundbreaking work on ImageNet
  - They did not have much experience on general image classification on ImageNet
  - It took one week to train the network with 60 Million parameters
  - The learned feature representations are effective on other datasets (e.g. Pascal VOC) and other tasks (object detection, segmentation, tracking, and image retrieval)



96 learned low-level filters

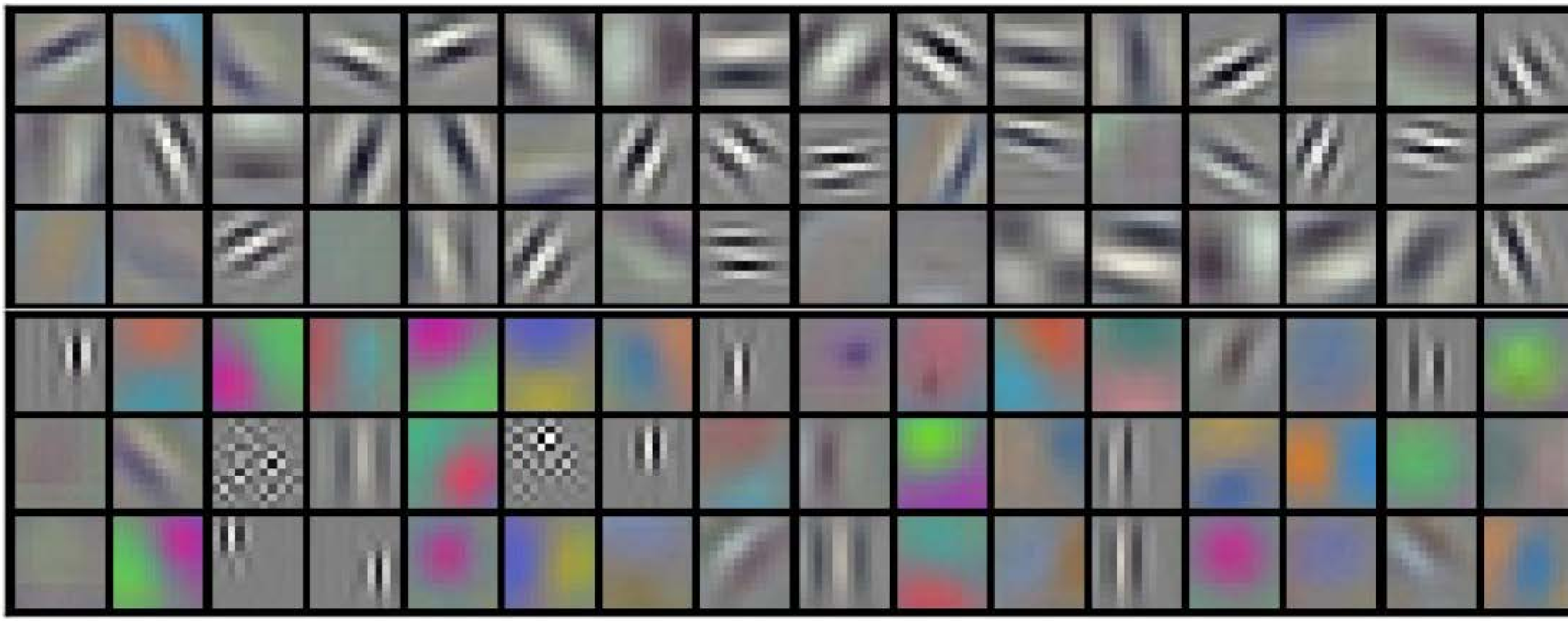
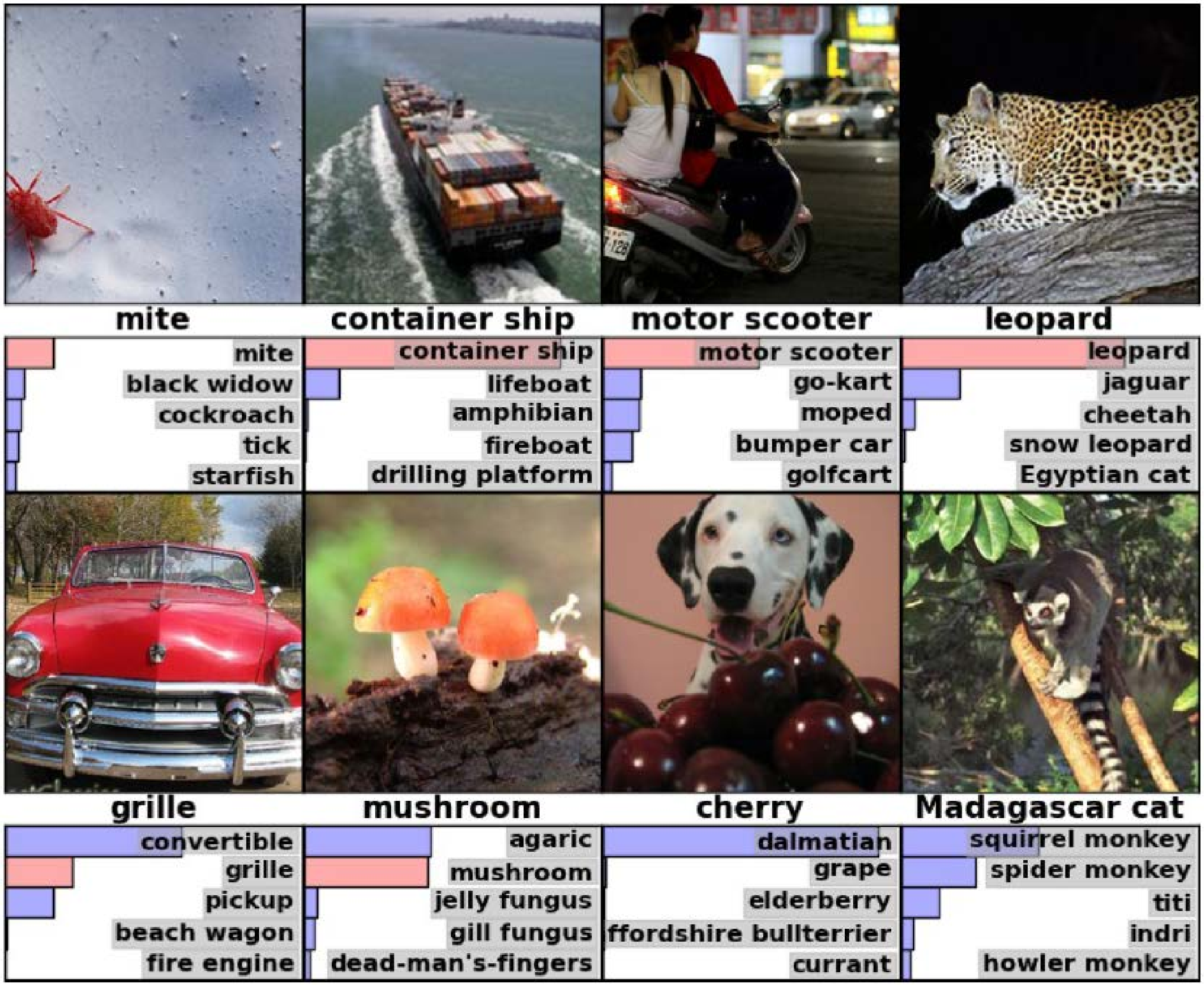
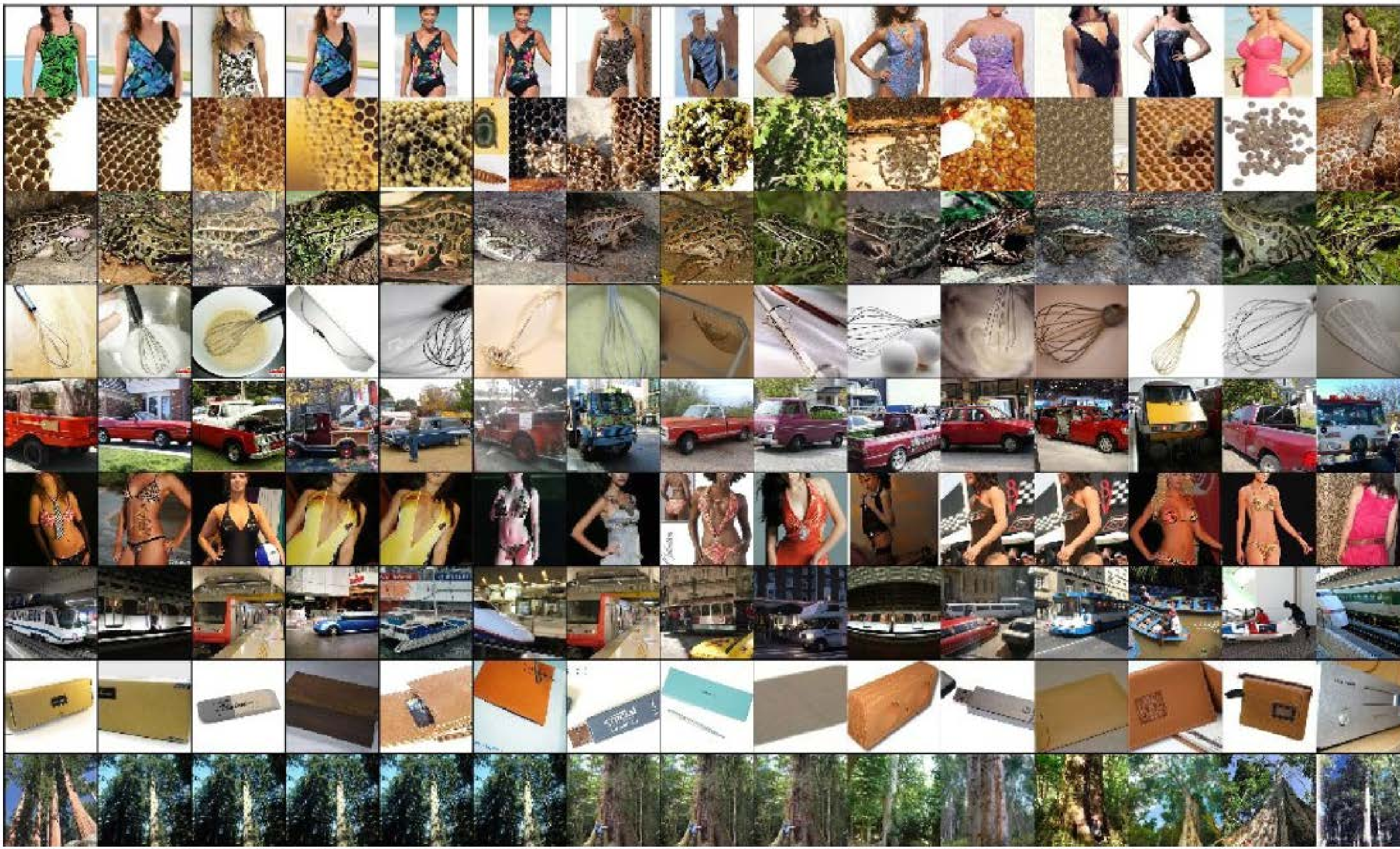


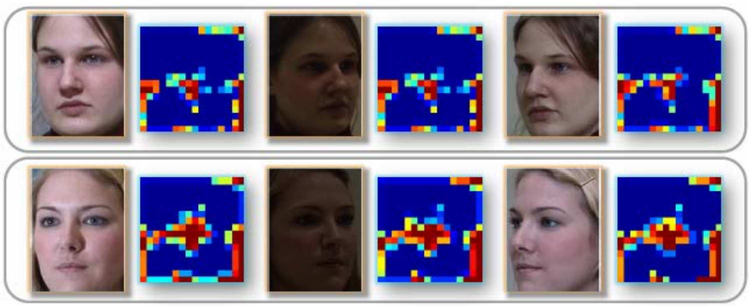
Image classification result

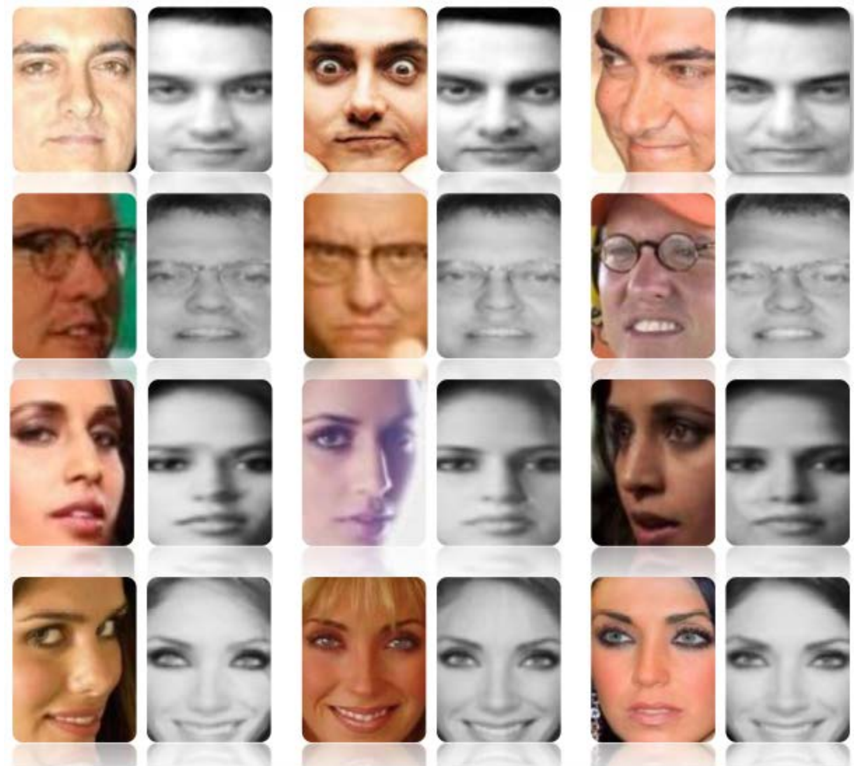


Top hidden layer can be used as feature for retrieval



Example 2: deep learning face identity features by recovering canonical-view face images

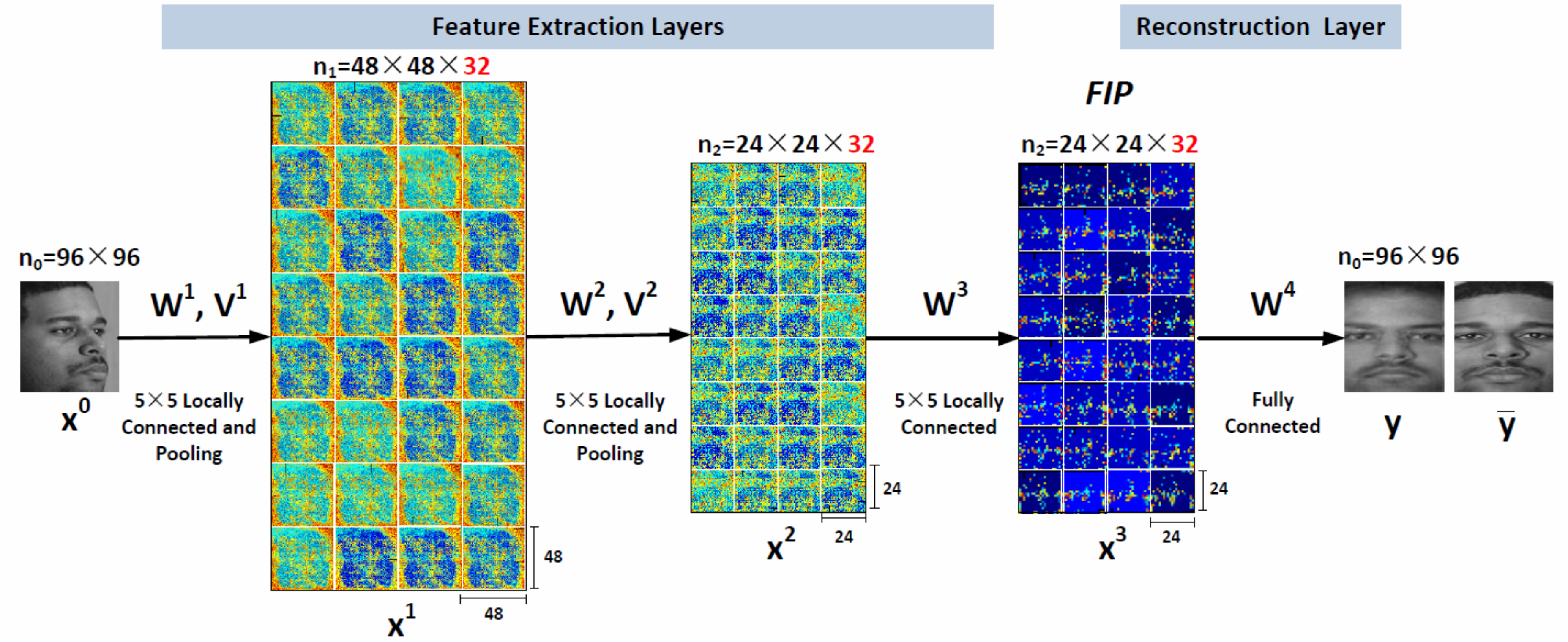




Reconstruction examples from LFW

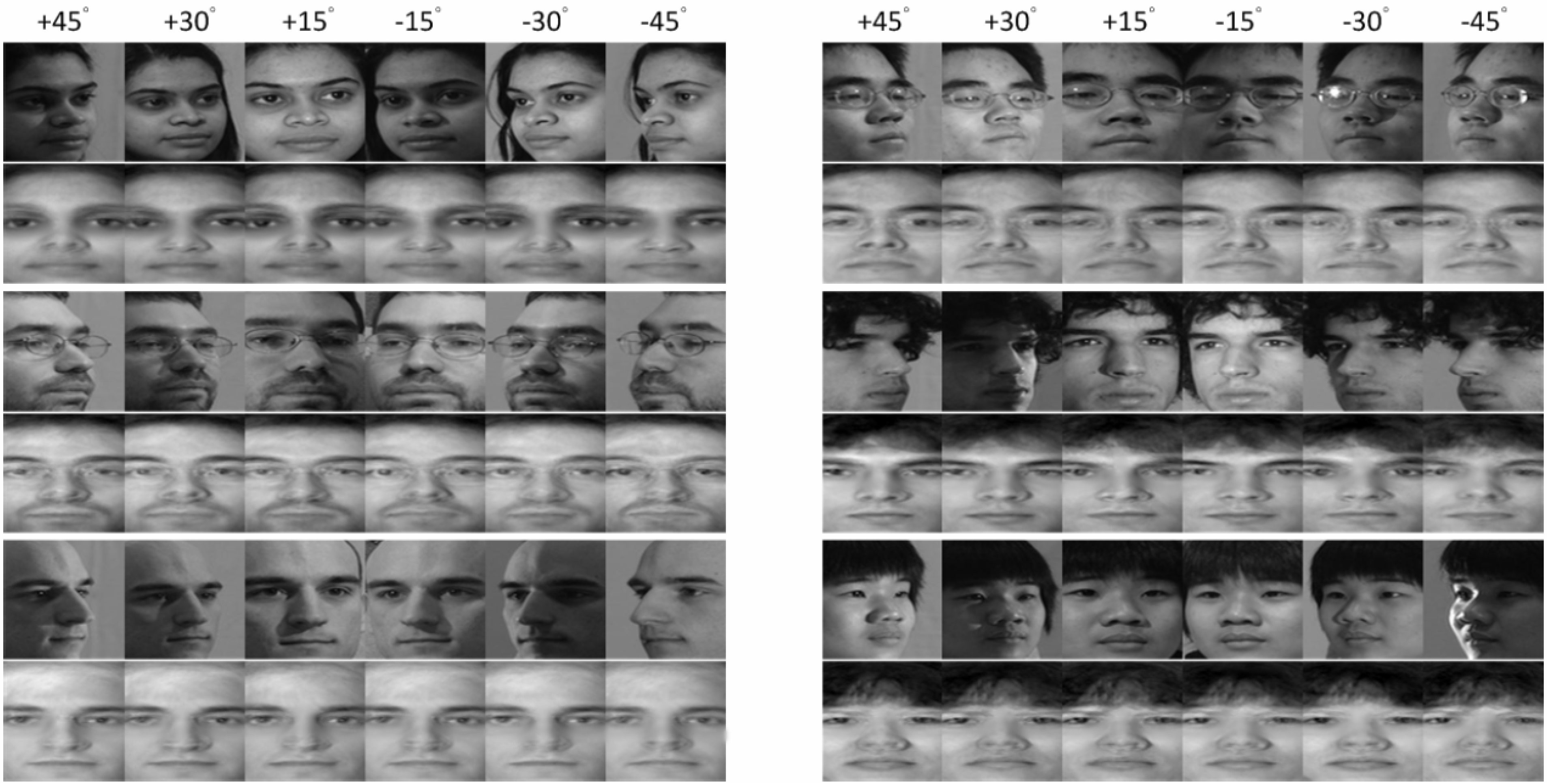
Z. Zhu, P. Luo, X. Wang, and X. Tang, "Deep Learning Identity Preserving Face Space," ICCV 2013.

- Deep model can disentangle hidden factors through feature extraction over multiple layers
- No 3D model; no prior information on pose and lighting condition
- Model multiple complex transforms
- Reconstructing the whole face is a much strong supervision than predicting 0/1 class label and helps to avoid overfitting



Arbitrary view

Canonical view


+45° +30° +15° -15° -30° -45°
+45° +30° +15° -15° -30° -45°

Comparison on Multi-PIE

	-45°	-30°	-15°	+15°	+30°	+45°	Avg	Pose
LGBP [26]	37.7	62.5	77	83	59.2	36.1	59.3	√
VAAM [17]	74.1	91	95.7	95.7	89.5	74.8	86.9	√
FA-EGFC[3]	84.7	95	99.3	99	92.9	85.2	92.7	x
SA-EGFC[3]	93	**98.7**	99.7	**99.7**	**98.3**	93.6	97.2	√
LE[4] + LDA	86.9	95.5	99.9	**99.7**	95.5	81.8	93.2	x
CRBM[9] + LDA	80.3	90.5	94.9	96.4	88.3	89.8	87.6	x
Ours	**95.6**	**98.5**	**100.0**	**99.3**	**98.5**	**97.8**	**98.3**	x

[3] A. Asthana, T. K. Marks, M. J. Jones, K. H. Tieu, and M. Rohith. Fully automatic pose-invariant face recognition via 3d pose normalization. In *ICCV*, pages 937–944, 2011. 1, 5, 6

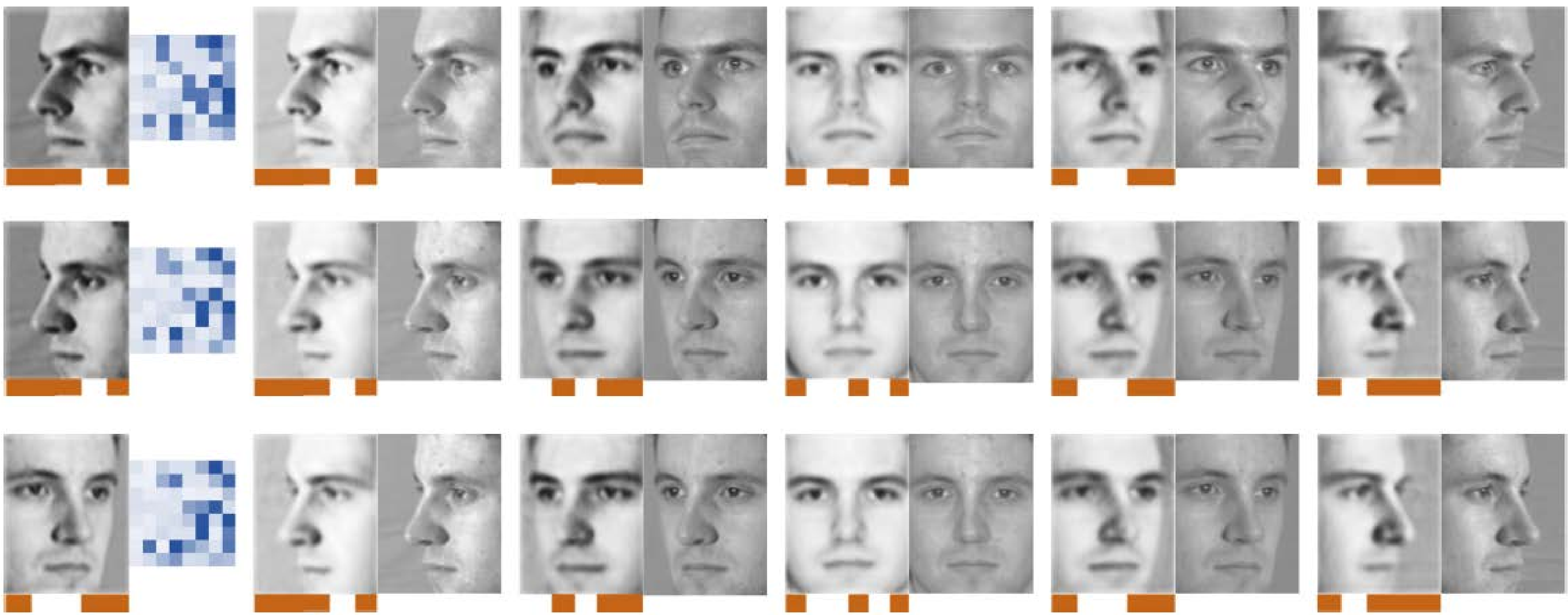
[4] Z. Cao, Q. Yin, X. Tang, and J. Sun. Face recognition with learning-based descriptor. In *CVPR*, pages 2707–2714, 2010. 2, 3, 6

[9] G. B. Huang, H. Lee, and E. Learned-Miller. Learning hierarchical representations for face verification with convolutional deep belief networks. In *CVPR*, pages 2518–2525, 2012. 3, 6

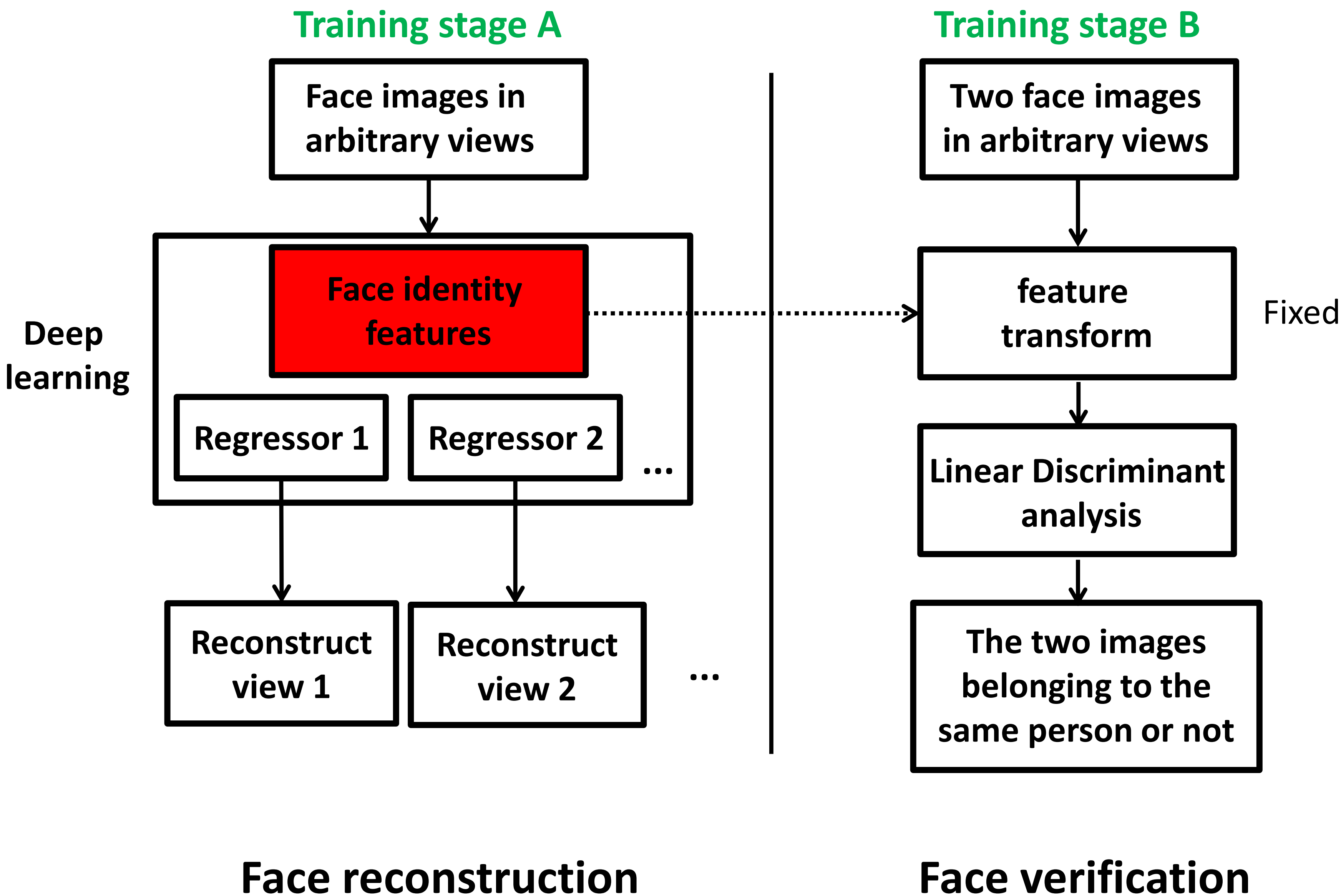
[17] S. Li, X. Liu, X. Chai, H. Zhang, S. Lao, and S. Shan. Morphable displacement field based image matching for face recognition across pose. In *ECCV*, pages 102–115. 2012. 1, 2, 5, 6

[26] W. Zhang, S. Shan, W. Gao, X. Chen, and H. Zhang. Local gabor binary pattern histogram sequence (lgbphs): A novel non-statistical model for face representation and recognition. In *ICCV*, volume 1, pages 786–791, 2005. 5, 6

Deep learning 3D model from 2D images, mimicking human brain activities

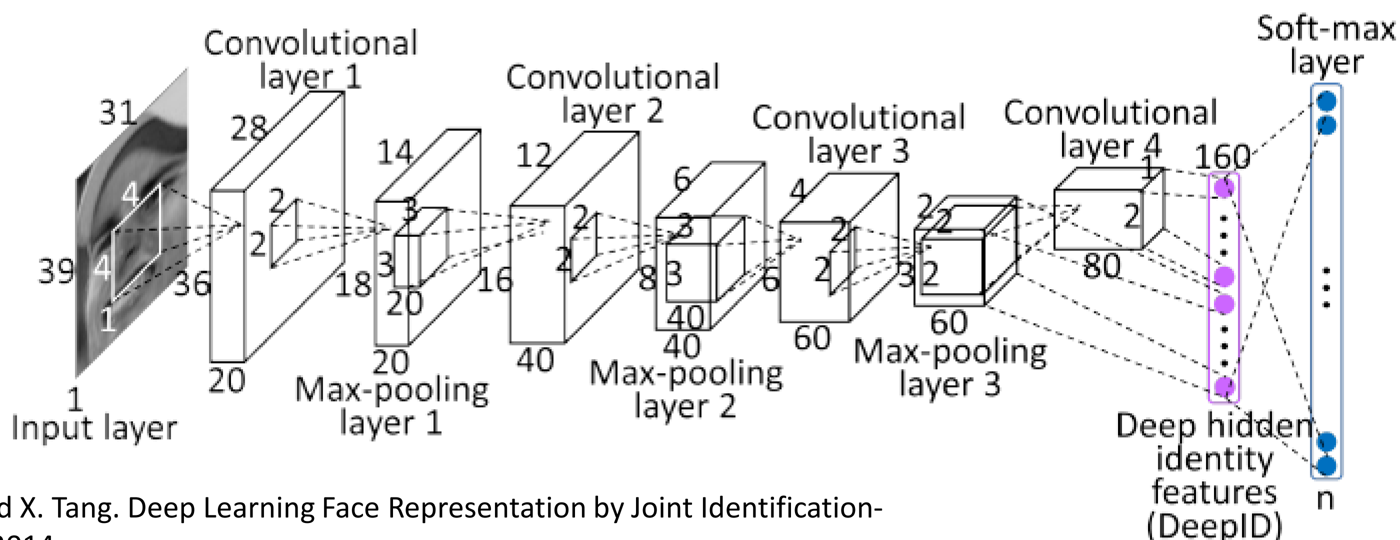


Z. Zhu, P. Luo, X. Wang, and X. Tang, “Deep Learning and Disentangling Face Representation by Multi-View Perception,” NIPS 2014.

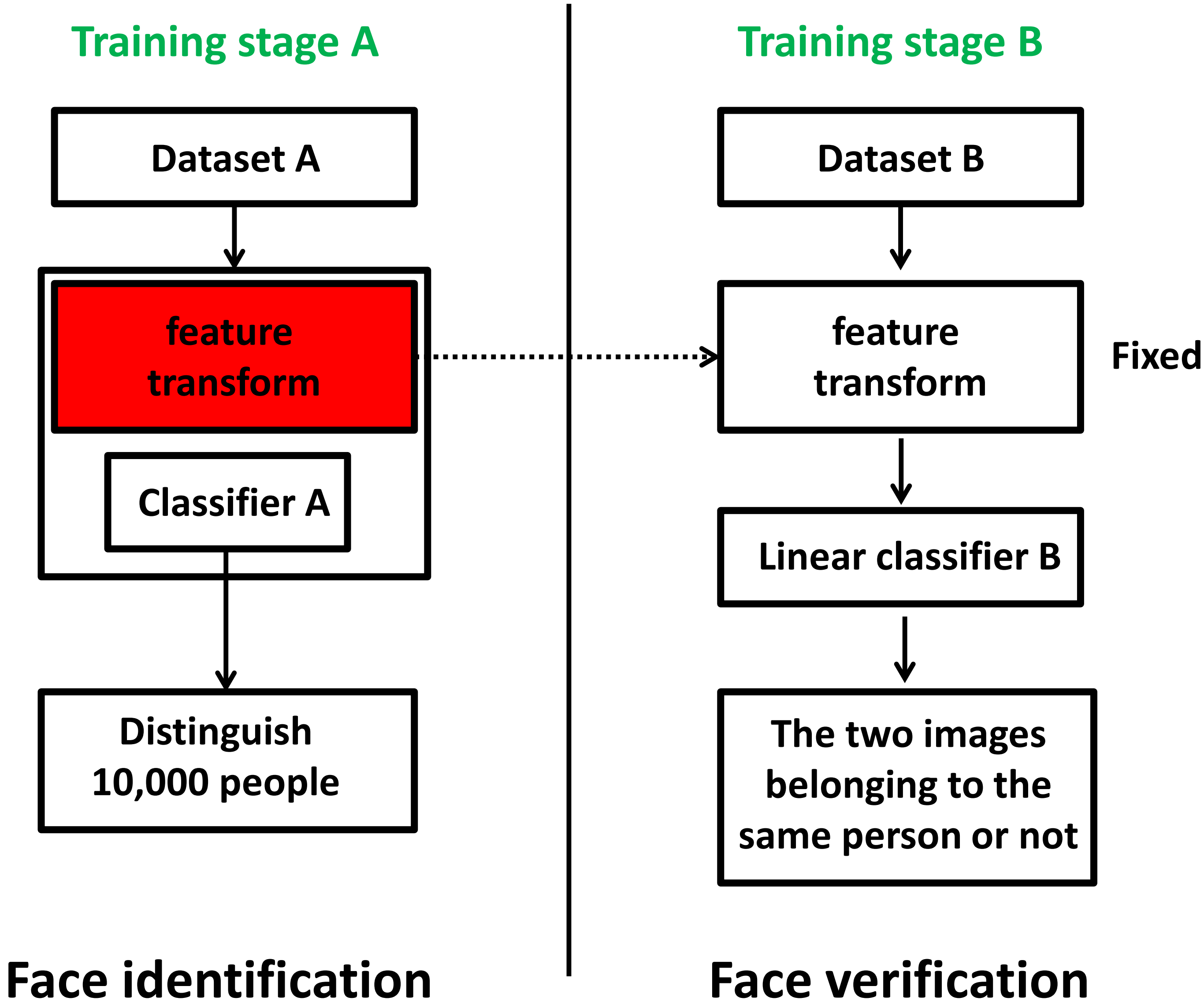

Training stage A
Face images in arbitrary views
Deep learning
Face identity features
Regressor 1
Regressor 2
...
Reconstruct view 1
Reconstruct view 2
...
Face reconstruction
Training stage B
Two face images in arbitrary views
feature transform
Fixed
Linear Discriminant analysis
The two images belonging to the same person or not
Face verification

Example 3: deep learning face identity features from predicting 10,000 classes

- At training stage, each input image is classified into 10,000 identities with 160 hidden identity features in the top layer
- The hidden identity features can be well generalized to other tasks (e.g. verification) and identities outside the training set
- As adding the number of classes to be predicted, the generalization power of the learned features also improves



Y. Sun, X. Wang, and X. Tang. Deep Learning Face Representation by Joint Identification-Verification. NIPS, 2014.

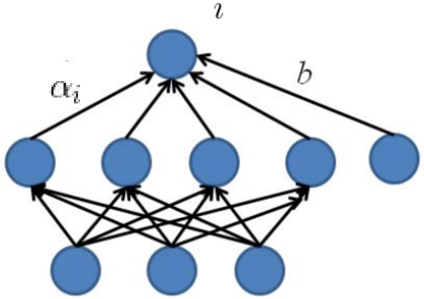

Training stage A
Dataset A
feature transform
Classifier A
Distinguish 10,000 people
Face identification
Training stage B
Dataset B
feature transform
Fixed
Linear classifier B
The two images belonging to the same person or not
Face verification

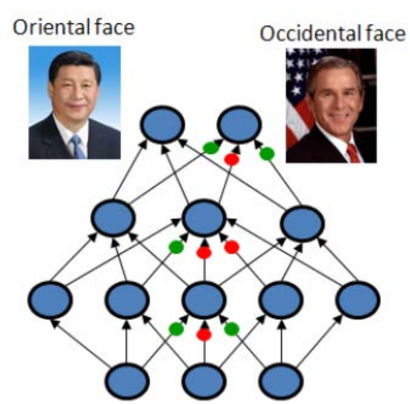
Deep Structures vs Shallow Structures (Why deep?)

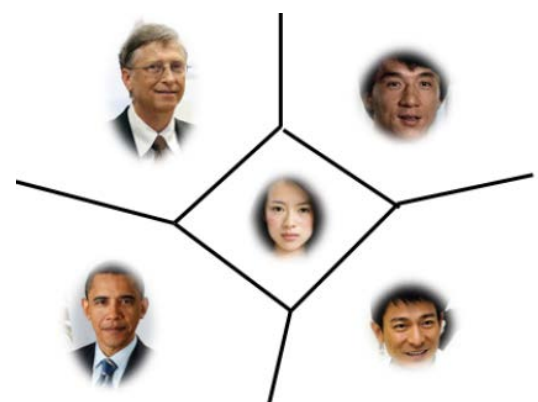
Shallow Structures

- A three-layer neural network (with one hidden layer) can approximate any classification function
- Most machine learning tools (such as SVM, boosting, and KNN) can be approximated as neural networks with one or two hidden layers
- Shallow models divide the feature space into regions and match templates in local regions. O(N) parameters are needed to represent N regions

SVM $g(x) = b + \sum_i \alpha_i K(x, x_i)$







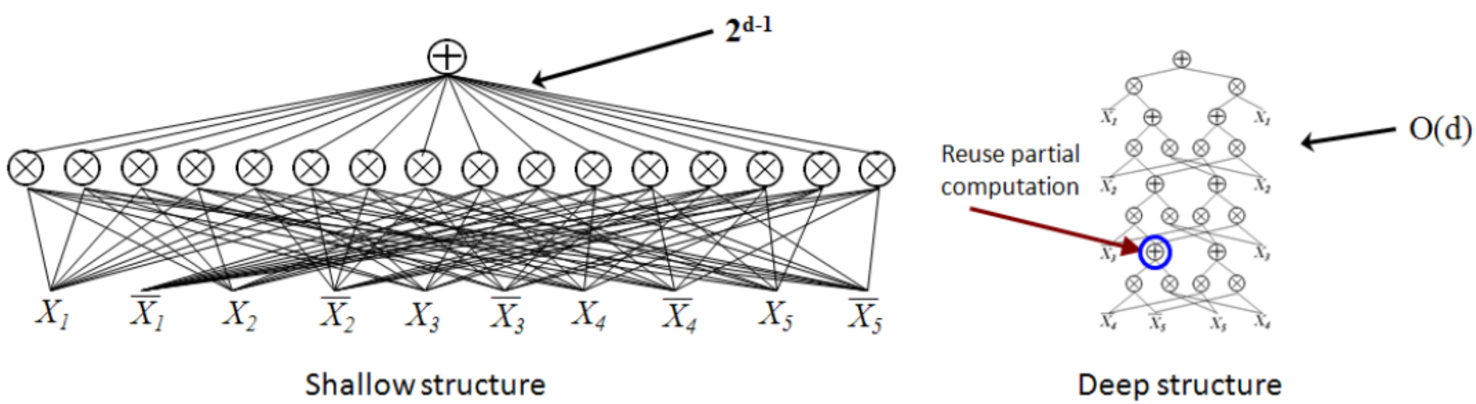
Deep Machines are More Efficient for Representing Certain Classes of Functions

- Theoretical results show that an architecture with insufficient depth can require many more computational elements, potentially exponentially more (with respect to input size), than architectures whose **depth is matched to the task** (Hastad 1986, Hastad and Goldmann 1991)
- It also means many more parameters to learn

- Take the d-bit parity function as an example

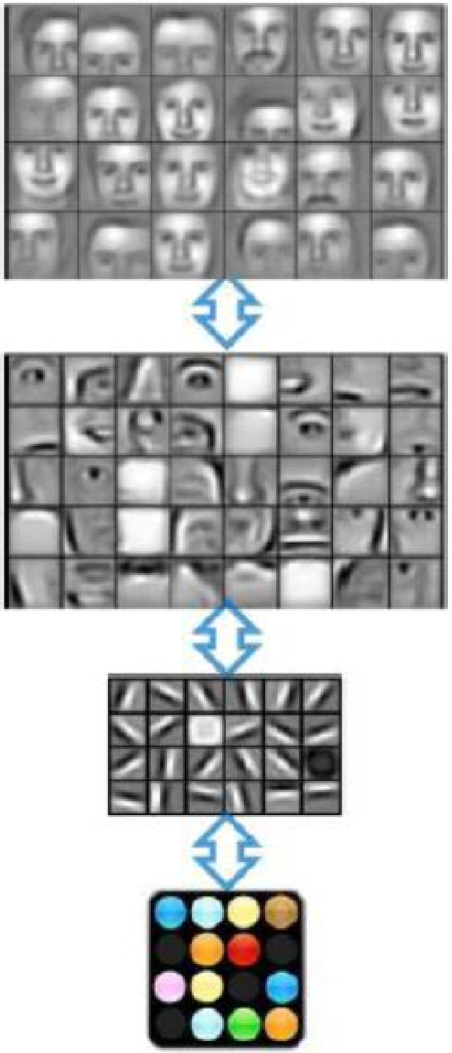
$$(X_{1,} \ldots , X_d) \in \{0,1\}^d \longmapsto \begin{cases} 1, & \text{if } \sum_{i=1}^{d} X_i \text{ is even} \\ -1, & \text{otherwise} \end{cases}$$

- d-bit logical parity circuits of depth 2 have exponential size (Andrew Yao, 1985)



- There are functions computable with a polynomial-size logic gates circuits of depth k that require exponential size when restricted to depth k -1 (Hastad, 1986)

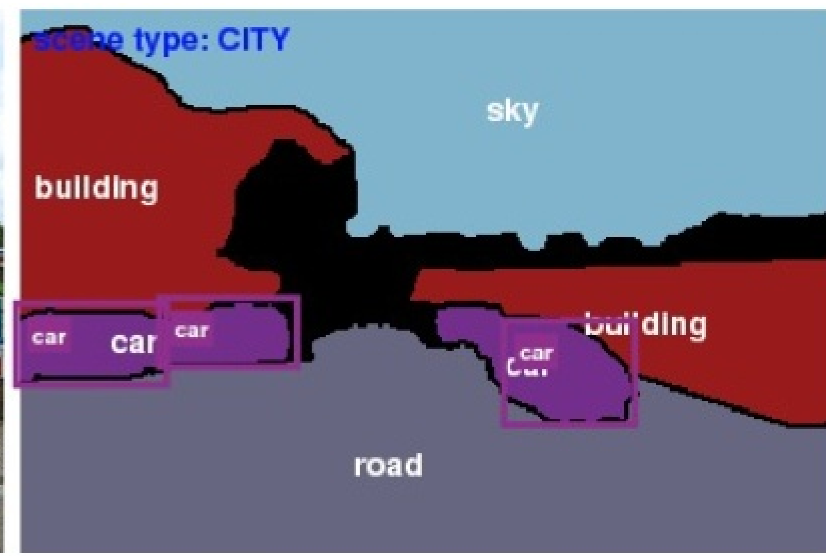
- Architectures with multiple levels naturally provide sharing and re-use of components



Honglak Lee, NIPS'10

Humans Understand the World through Multiple Levels of Abstractions

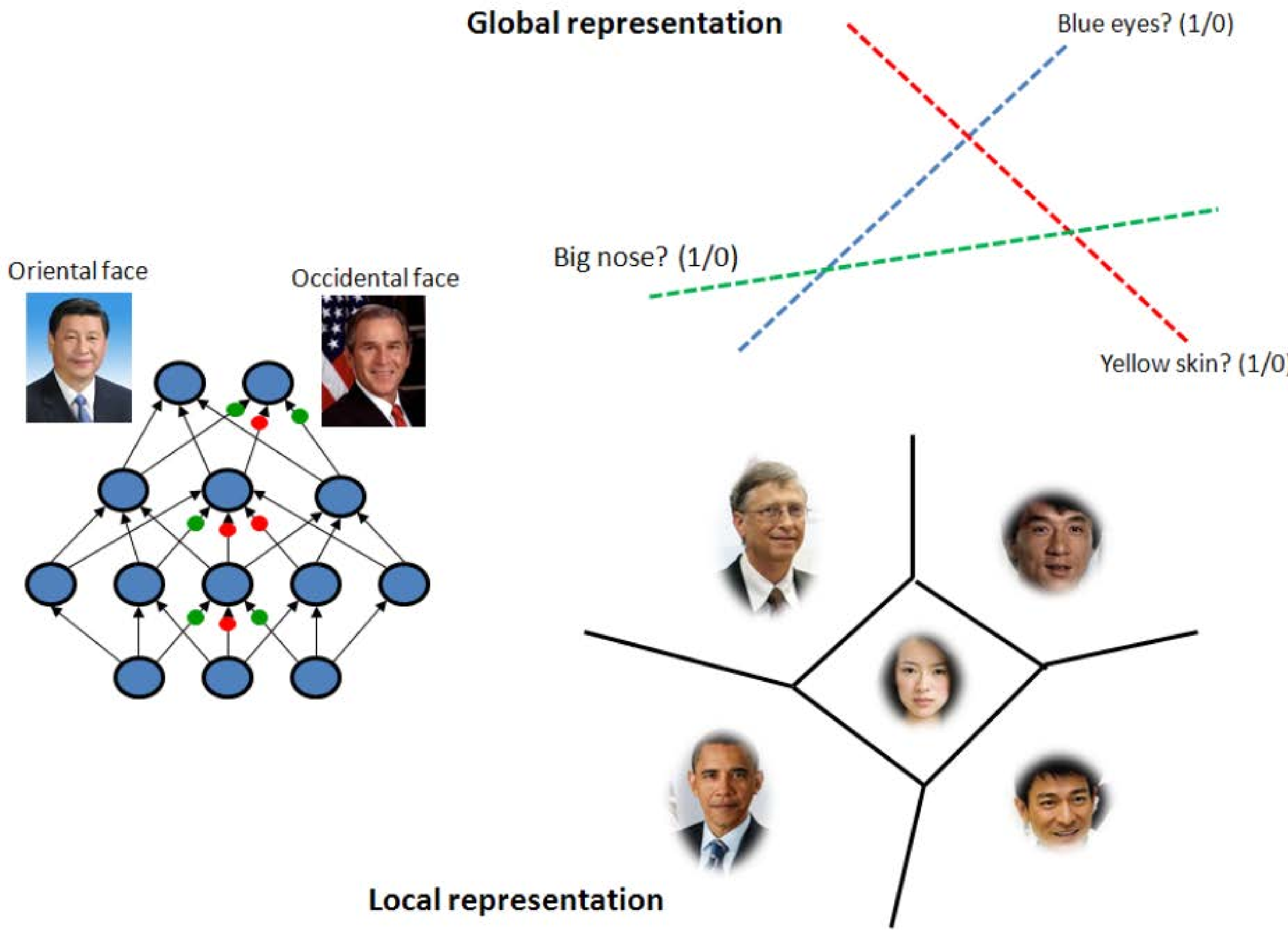
- We do not interpret a scene image with pixels
  - Objects (sky, cars, roads, buildings, pedestrians) -> parts (wheels, doors, heads) -> texture -> edges -> pixels
  - Attributes: blue sky, red car
- It is natural for humans to decompose a complex problem into sub-problems through multiple levels of representations



Humans Understand the World through Multiple Levels of Abstractions

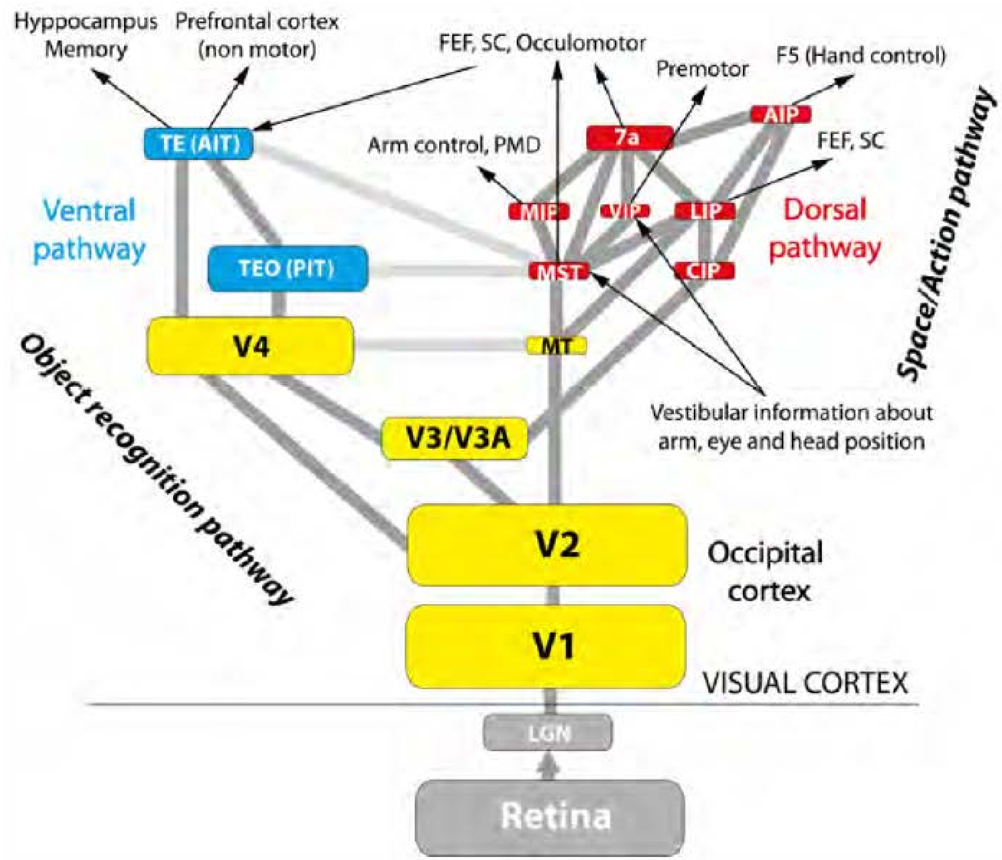
- Humans learn abstract concepts on top of less abstract ones
- Humans can imagine new pictures by re-configuring these abstractions at multiple levels. Thus our brain has good generalization can recognize things never seen before.
  - Our brain can estimate shape, lighting and pose from a face image and generate new images under various lightings and poses. That's why we have good face recognition capability.

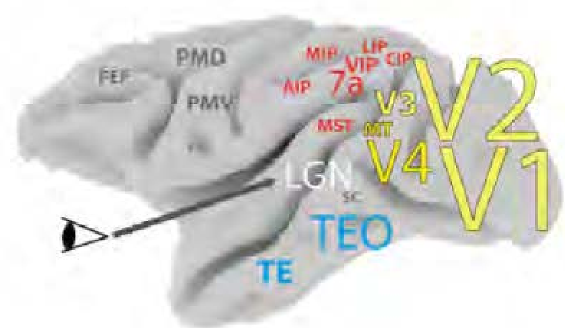
Local and Global Representations



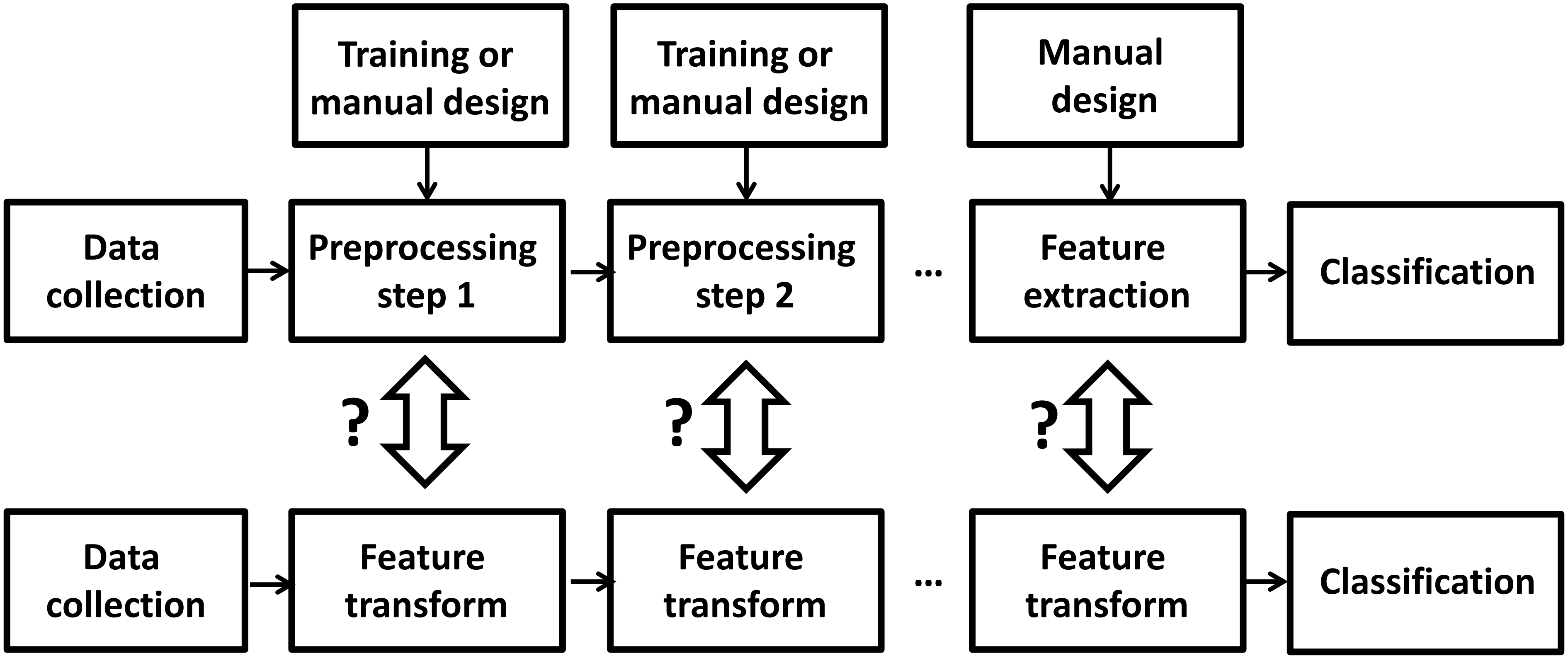
Human Brains Process Visual Signals through Multiple Layers

- A visual cortical area consists of six layers (Kruger et al. 2013)





Joint Learning vs Separate Learning



End-to-end learning

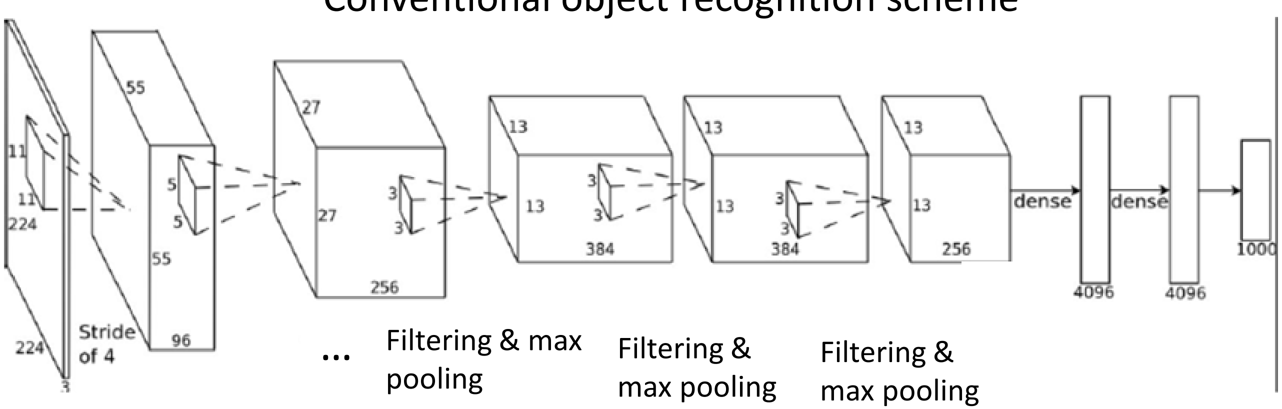
Deep learning is a framework/language but not a black-box model

Its power comes from joint optimization and increasing the capacity of the learner

- Domain knowledge could be helpful for designing new deep models and training strategies
- How to formulate a vision problem with deep learning?
  - Make use of experience and insights obtained in CV research
  - Sequential design/learning vs **joint learning**
  - Effectively train a deep model (layerwise pre-training + fine tuning)

Feature extraction → Quantization (visual words) → Spatial pyramid (histograms in local regions) → Classification

Conventional object recognition scheme



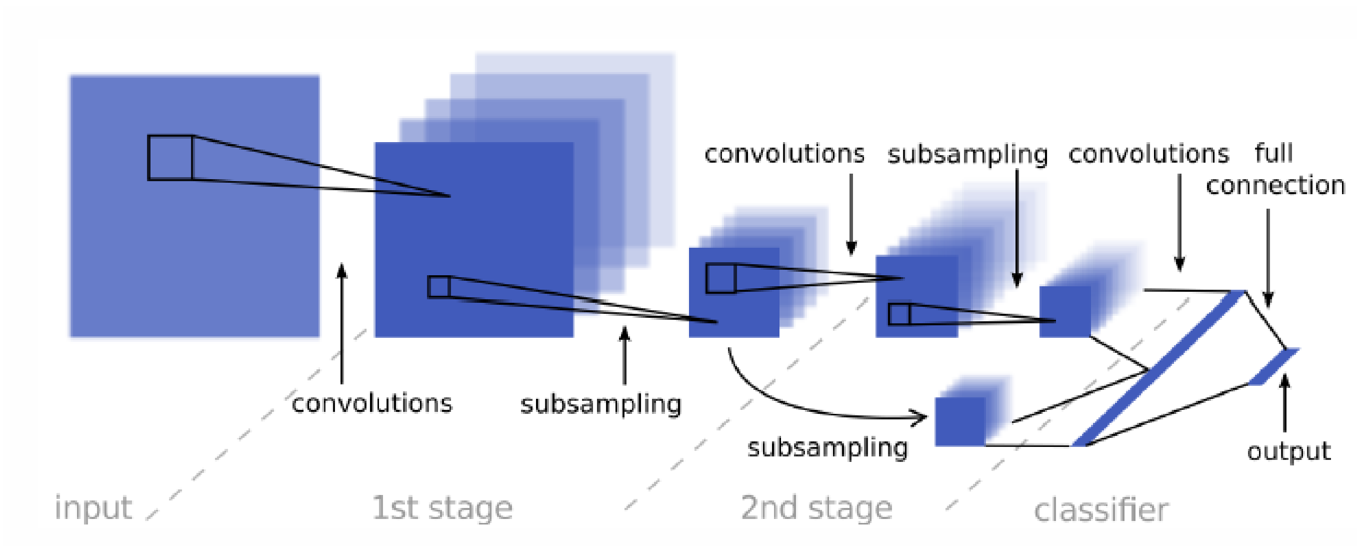
Feature extraction ↔ filtering

Quantization ↔ filtering

Spatial pyramid ↔ multi-level pooling

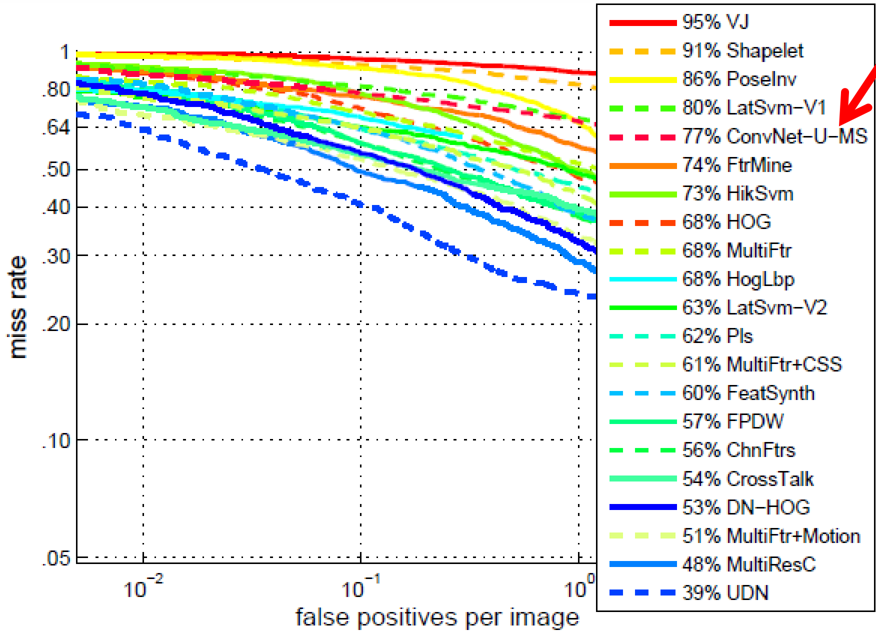
Krizhevsky NIPS'12

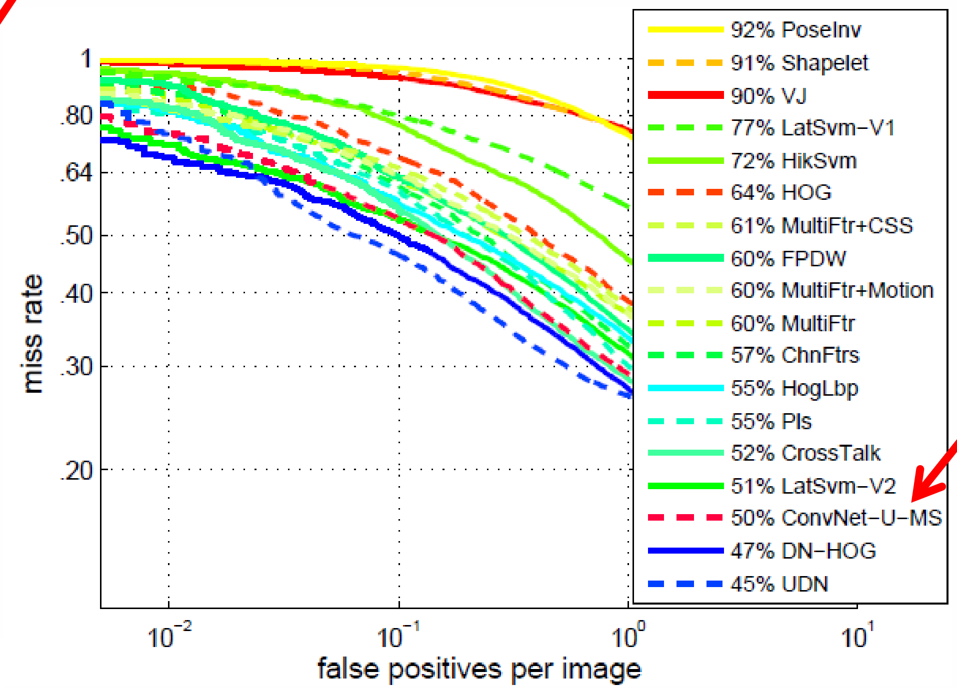
What if we treat an existing deep model as a black box in pedestrian detection?



ConvNet−U−MS

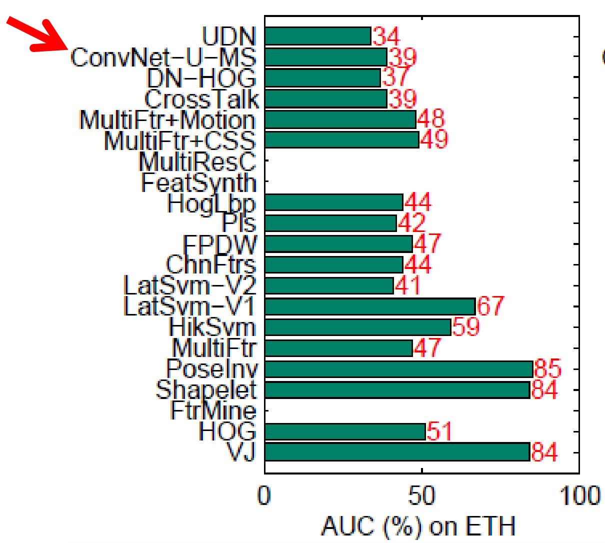
– Sermnet, K. Kavukcuoglu, S. Chintala, and LeCun, “Pedestrian Detection with Unsupervised Multi-Stage Feature Learning,” CVPR 2013.

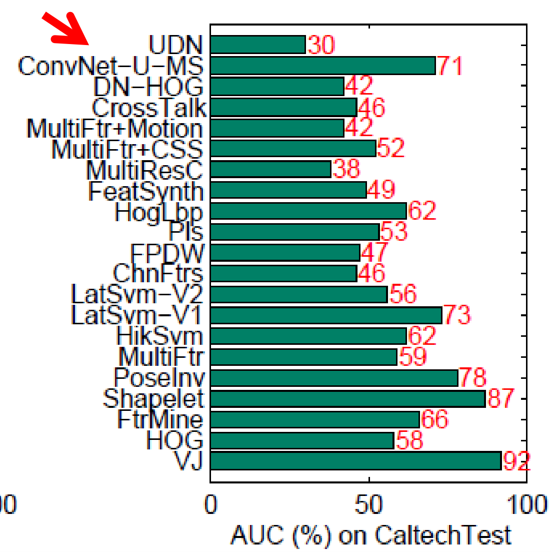


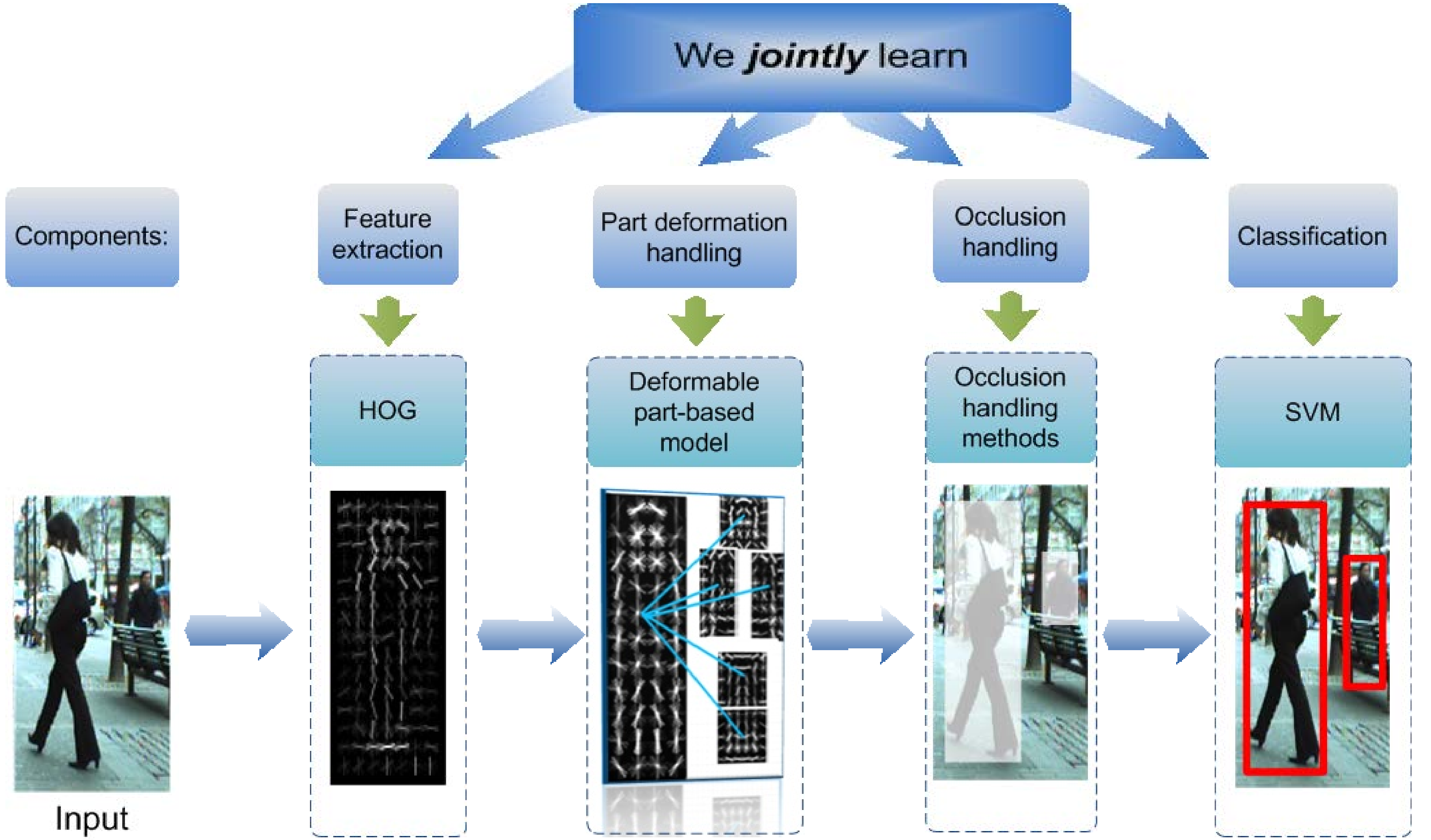


Results on Caltech Test

Results on ETHZ

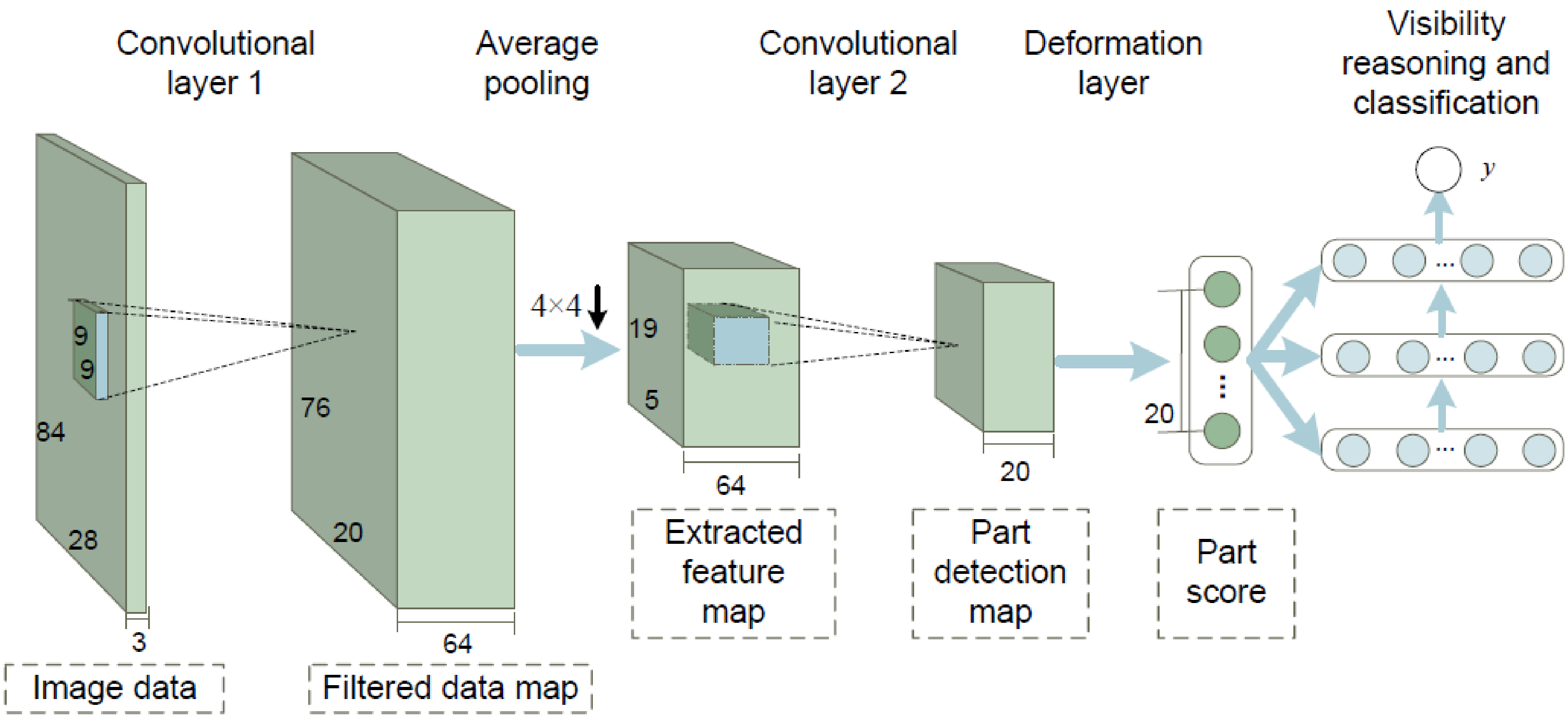






- N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. CVPR, 2005. (6000 citations)
- P. Felzenszwalb, D. McAlester, and D. Ramanan. A Discriminatively Trained, Multiscale, Deformable Part Model. CVPR, 2008. (2000 citations)
- W. Ouyang and X. Wang. A Discriminative Deep Model for Pedestrian Detection with Occlusion Handling. CVPR, 2012.

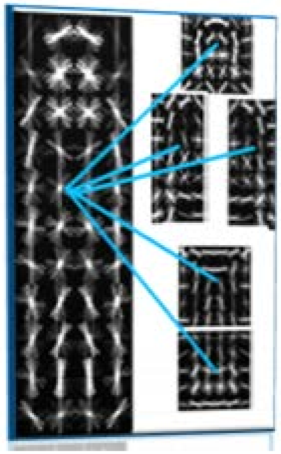
Our Joint Deep Learning Model



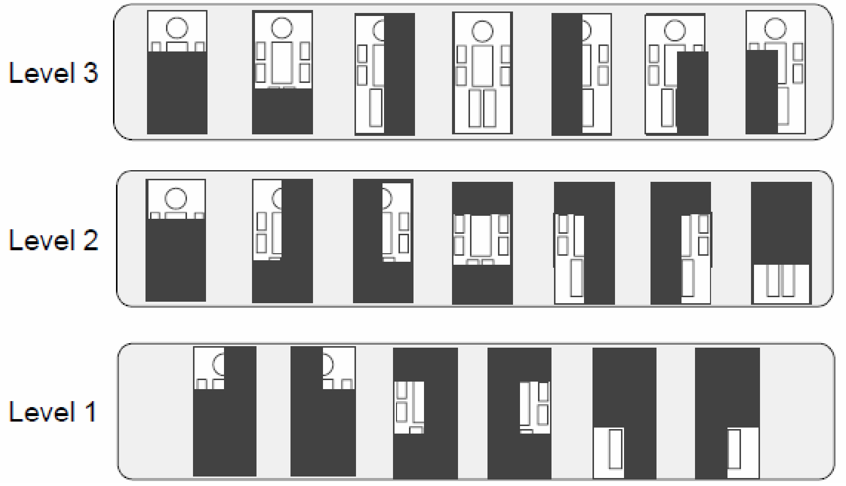
W. Ouyang and X. Wang, "Joint Deep Learning for Pedestrian Detection," Proc. ICCV, 2013.

Modeling Part Detectors

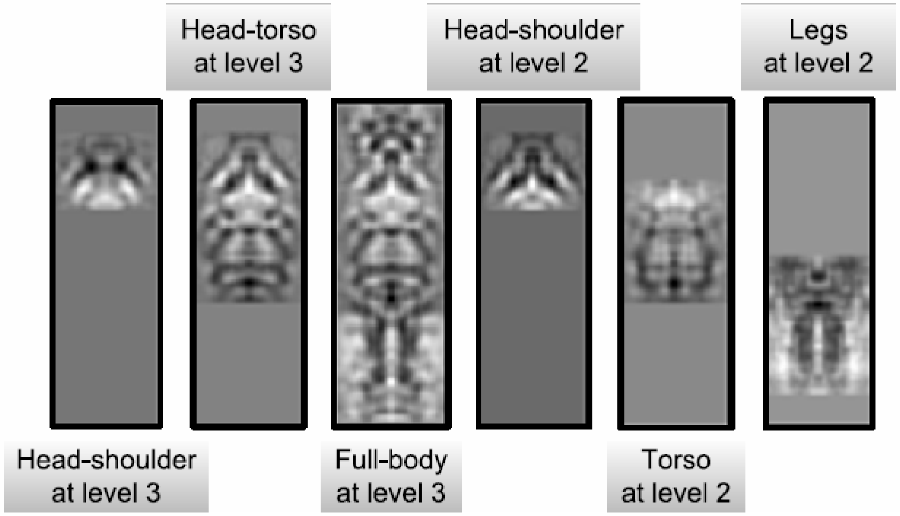
- Design the filters in the second convolutional layer with variable sizes



Part models learned from HOG

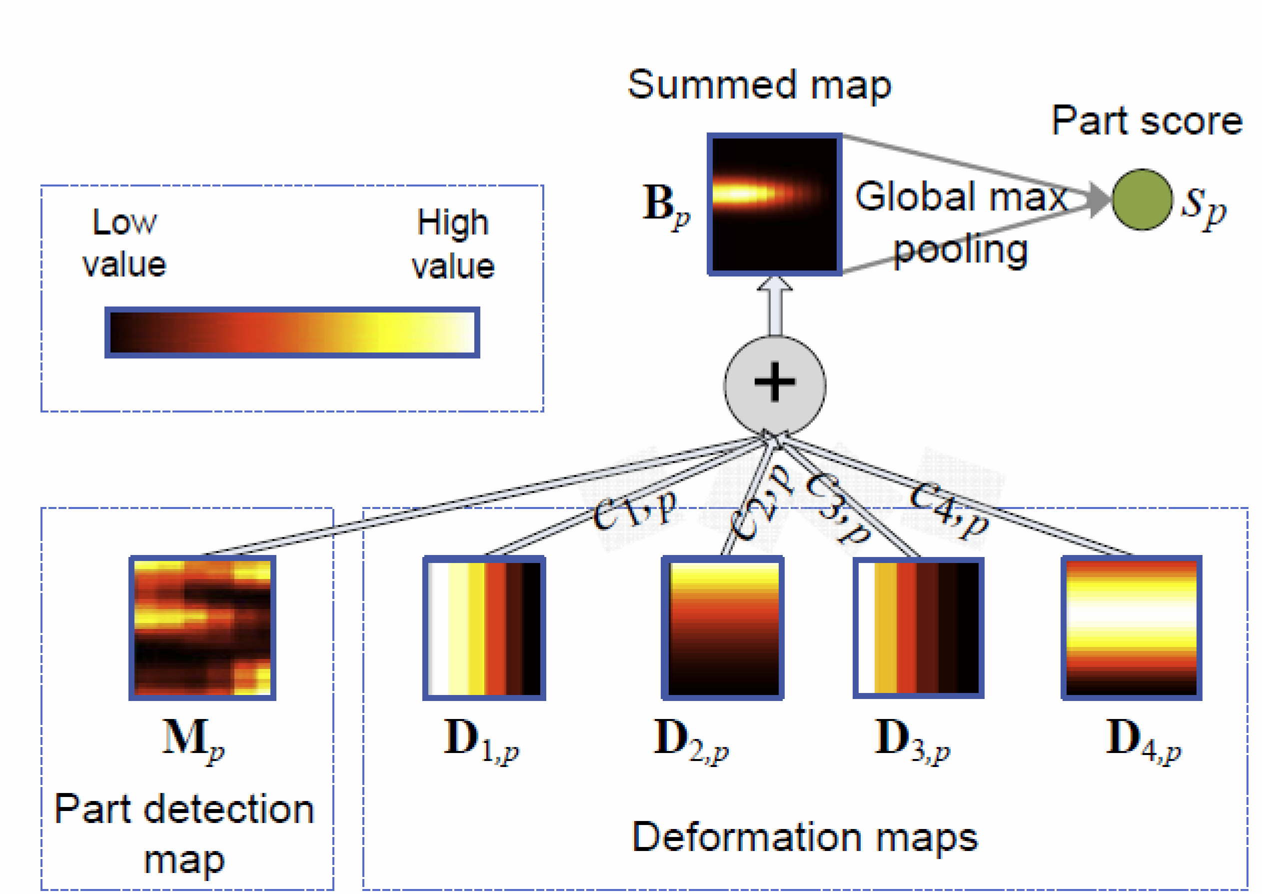


Part models

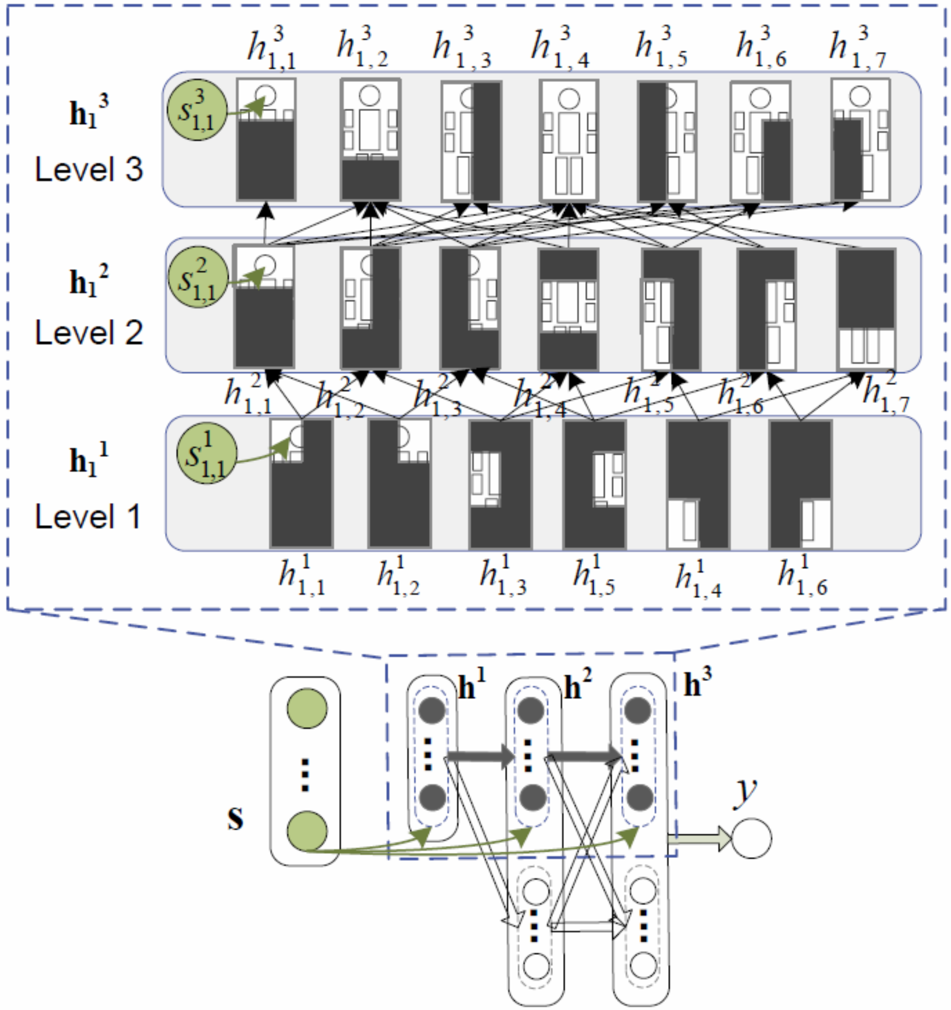


Learned filtered at the second convolutional layer

Deformation Layer



Visibility Reasoning with Deep Belief Net

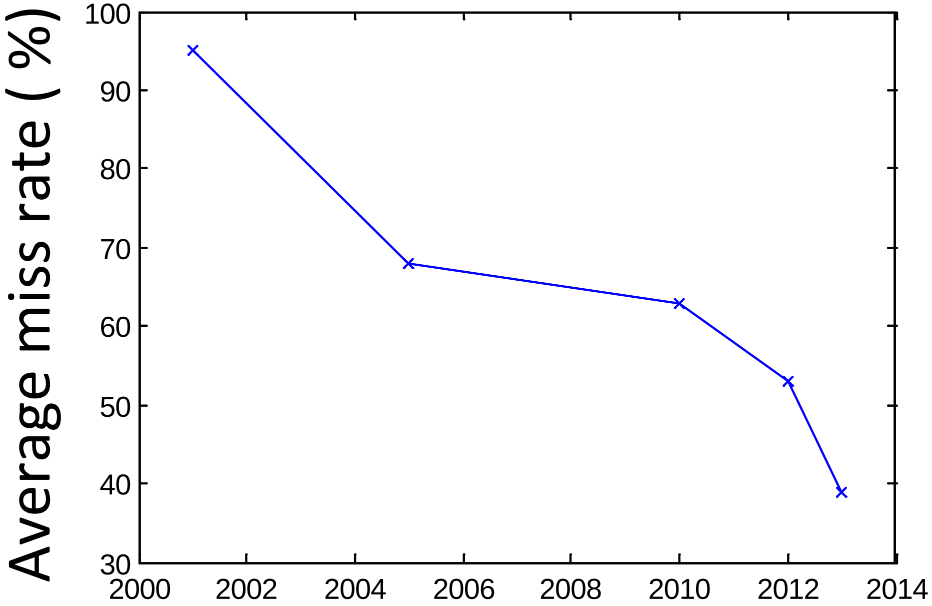


$$\tilde{h}_j^{l+1} = \sigma(\tilde{\mathbf{h}}^{l\mathrm{T}}\mathbf{w}_{*,j}^{l} + c_j^{l+1} + g_j^{l+1} s_j^{l+1})$$

Correlates with part detection score

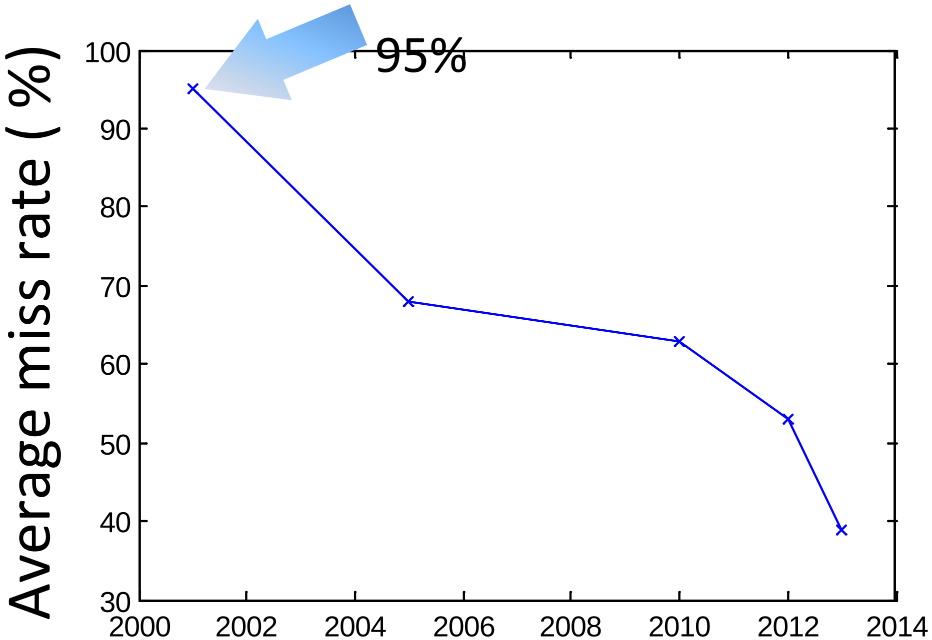
Experimental Results

- Caltech – Test dataset (largest, most widely used)



Experimental Results

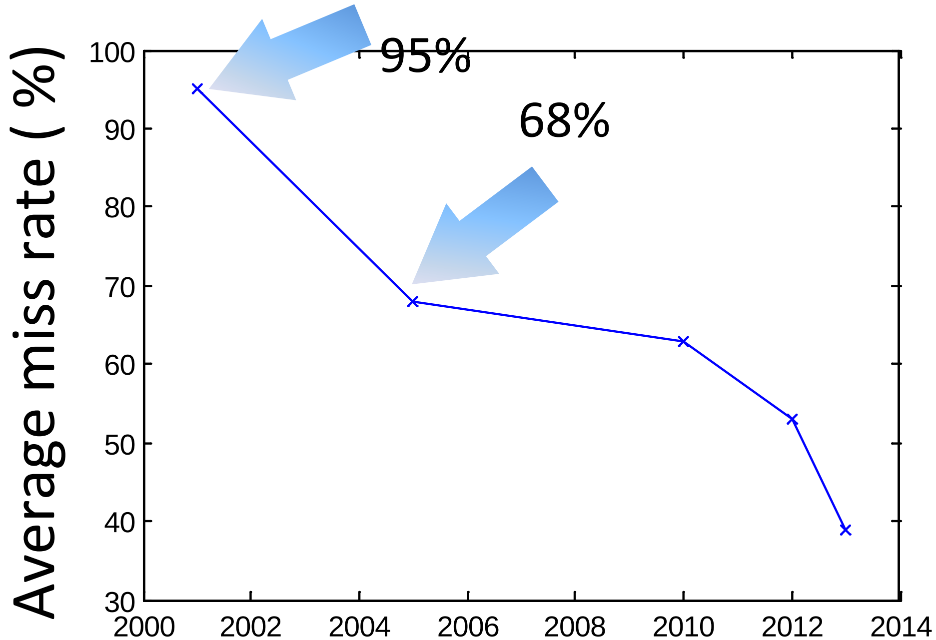
- Caltech – Test dataset (largest, most widely used)



Rapid object detection using a boosted cascade of simple features
P Viola, M Jones - … Vision and Pattern Recognition, 2001. CVPR …, 2001 - ieeexplore.ieee.org.org
Abstract This paper describes a machine learning approach for visual **object detection** which is capable of processing images extremely rapidly and achieving high **detection** rates. This work is distinguished by three key contributions. The first is the introduction of a new **...**
Cited by 7647 Related articles All 201 versions Import into BibTeX More

Experimental Results

- Caltech – Test dataset (largest, most widely used)



Histograms of oriented **gradients** for **human detection**

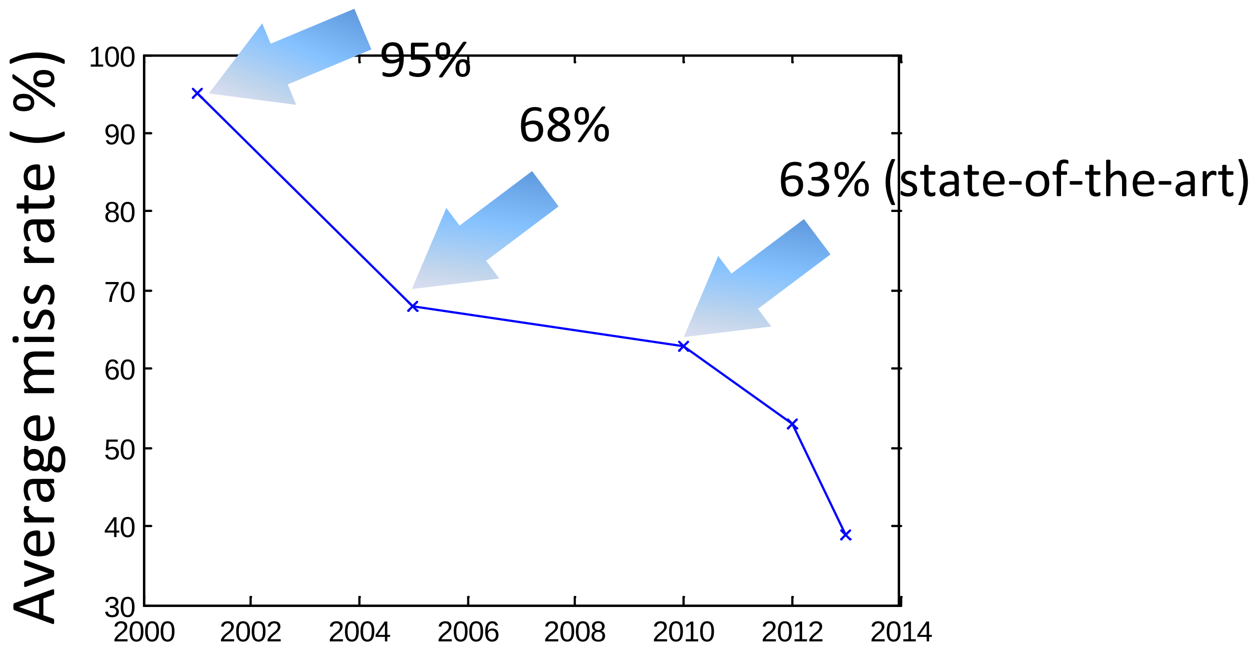
N Dalal, B Triggs - … and Pattern Recognition, 2005. CVPR 2005 …, 2005 - ieeexplore.ieee.org

... We study the issue of feature sets for **human detection**, showing that lo- cally normalized **Histogram** of Oriented **Gradient** (HOG) de- scriptors provide excellent performance relative to other ex- isting feature sets including wavelets [17,22]. **...**

Cited by 5438 Related articles All 106 versions Import into BibTeX More

Experimental Results

- Caltech – Test dataset (largest, most widely used)



Object detection with **discriminatively trained part-based models**

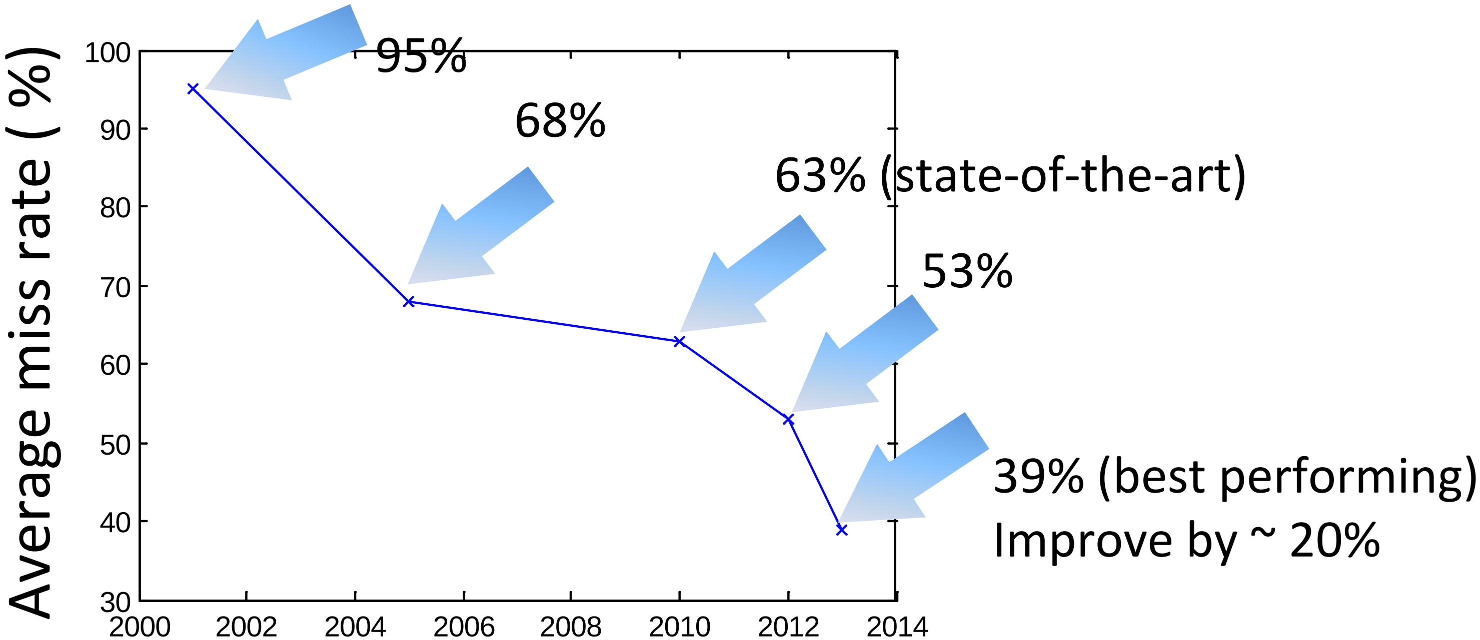
PF Felzenszwalb, RB Girshick… - Pattern Analysis and …, 2010 - ieeexplore.ieee.org

Abstract We describe an **object detection** system **based** on mixtures of multiscale deformable **part models**. Our system is able to represent highly variable **object** classes and achieves state-of-the-art results in the PASCAL **object detection** challenges. While **...**

Cited by 964 Related articles All 43 versions Import into BibTeX More

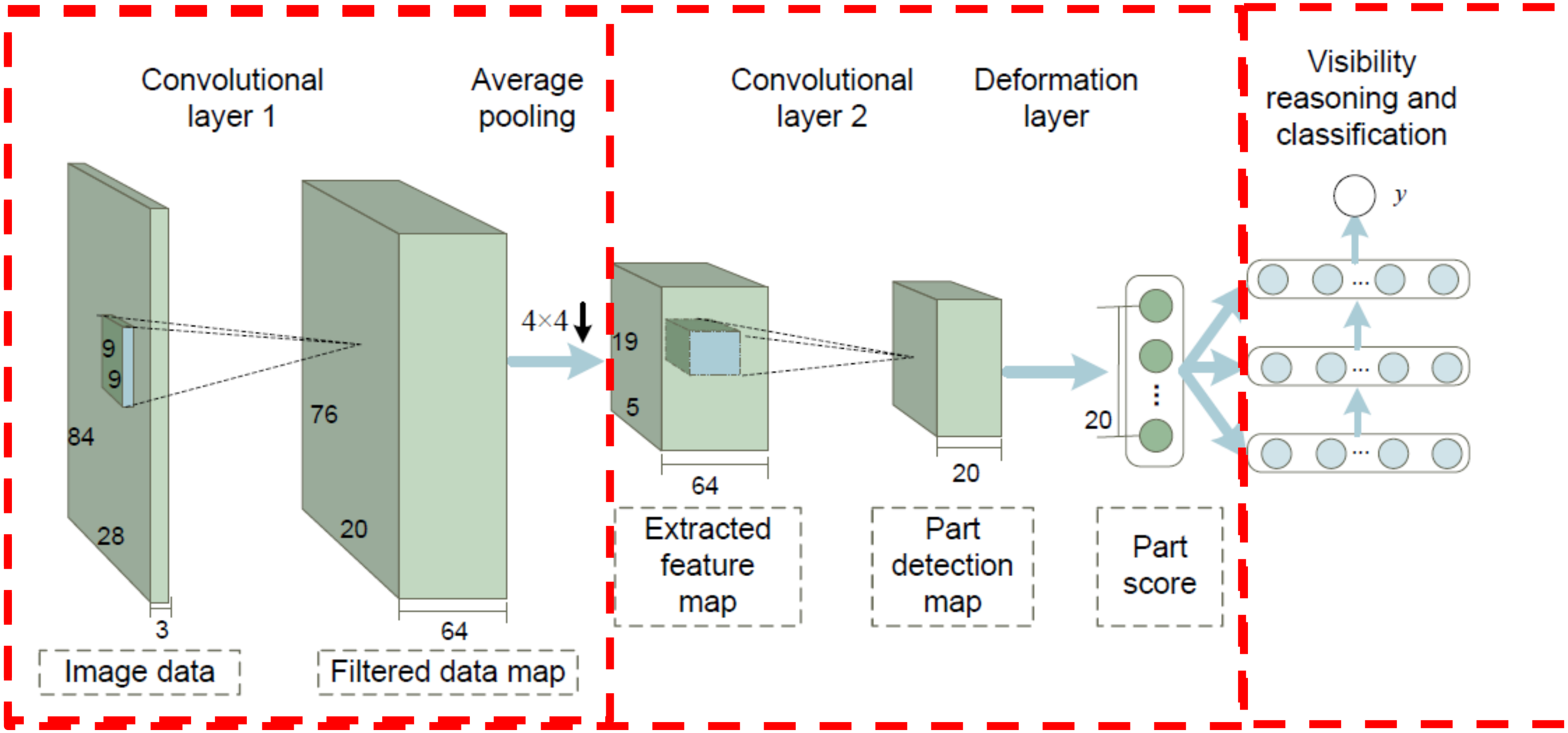
Experimental Results

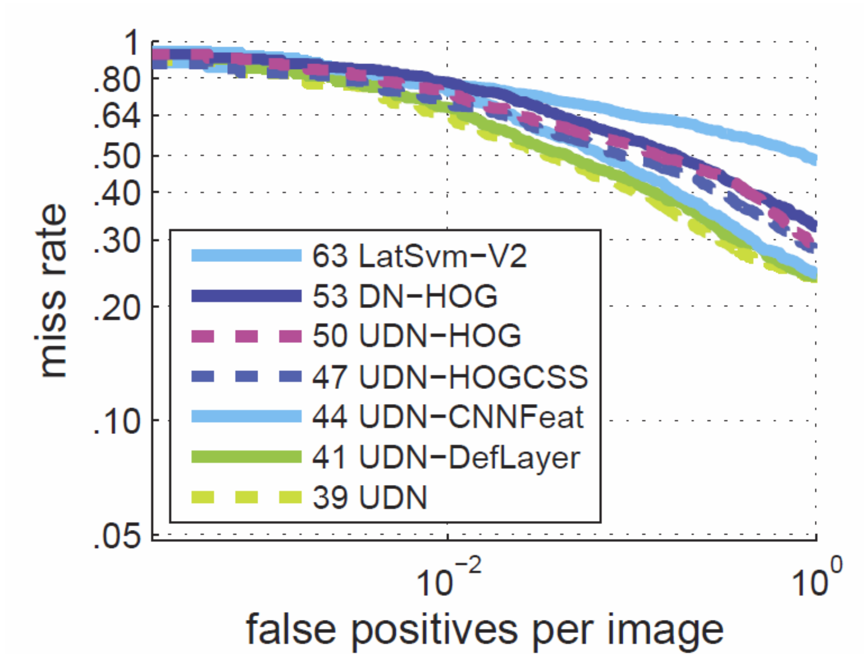
- Caltech – Test dataset (largest, most widely used)



W. Ouyang and X. Wang, "A Discriminative Deep Model for Pedestrian Detection with Occlusion Handling,“ CVPR 2012.

W. Ouyang, X. Zeng and X. Wang, "Modeling Mutual Visibility Relationship in Pedestrian Detection ", CVPR 2013.
W. Ouyang, Xiaogang Wang, "Single-Pedestrian Detection aided by Multi-pedestrian Detection ", CVPR 2013.
X. Zeng, W. Ouyang and X. Wang, ” A Cascaded Deep Learning Architecture for Pedestrian Detection,” ICCV 2013.
W. Ouyang and Xiaogang Wang, “Joint Deep Learning for Pedestrian Detection,” IEEE ICCV 2013.

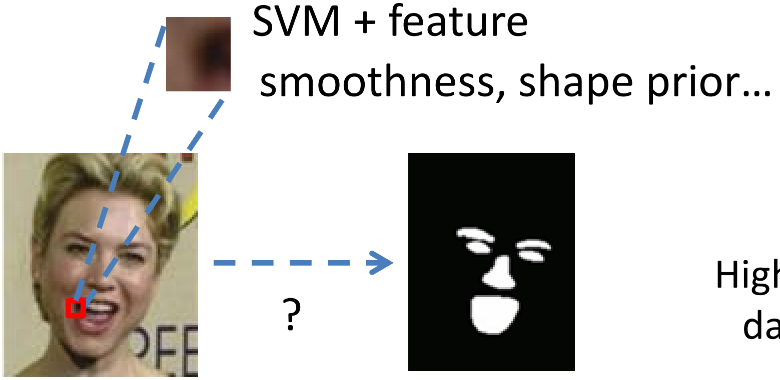


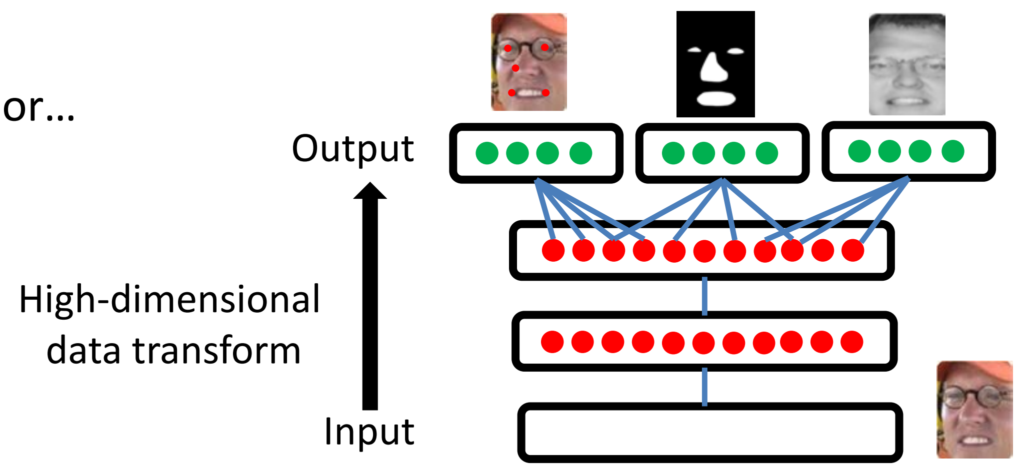


DN-HOG
UDN-HOG
UDN-HOGCSS
UDN-CNNFeat
UDN-DefLayer

Large learning capacity makes high dimensional data transforms possible, and makes better use of contextual information

- How to make use of the large learning capacity of deep models?
  - **High dimensional data transform**
  - Hierarchical nonlinear representations





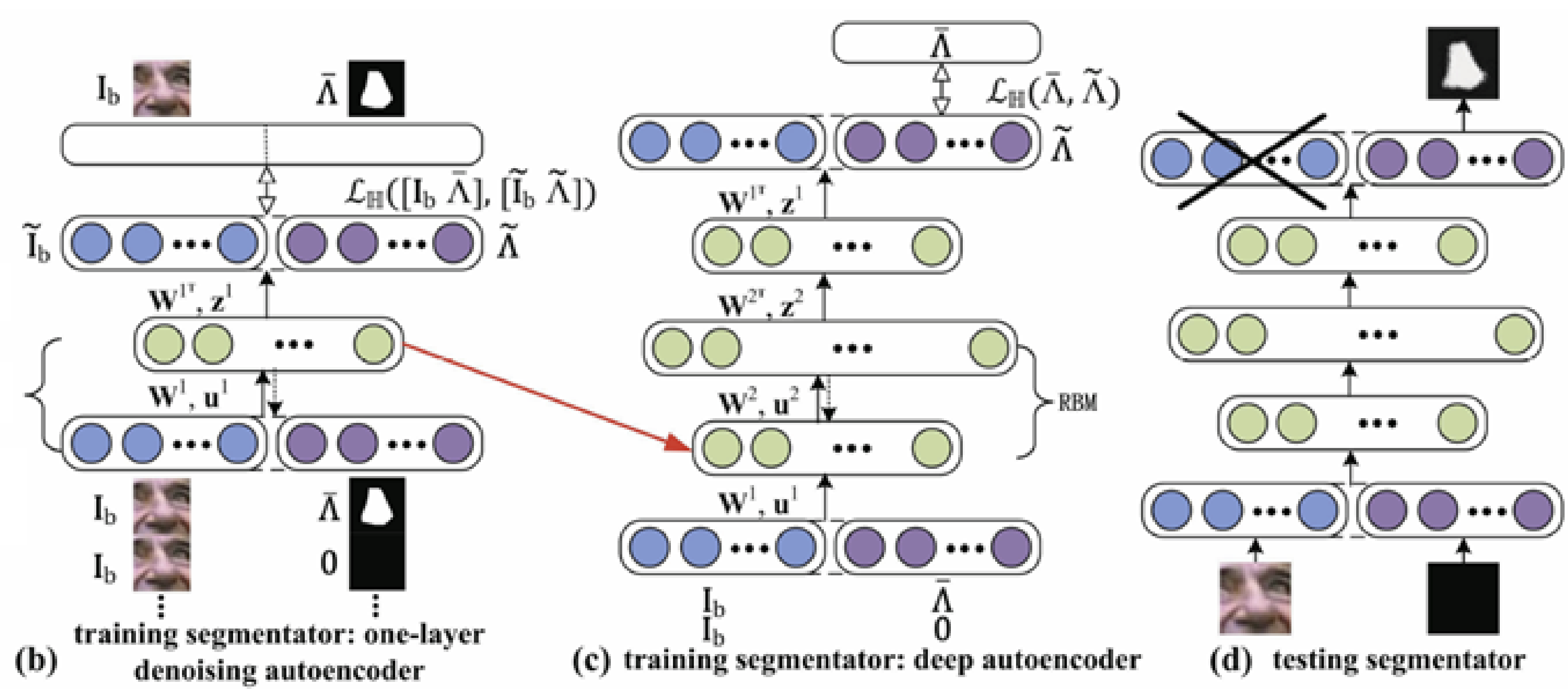
Face Parsing

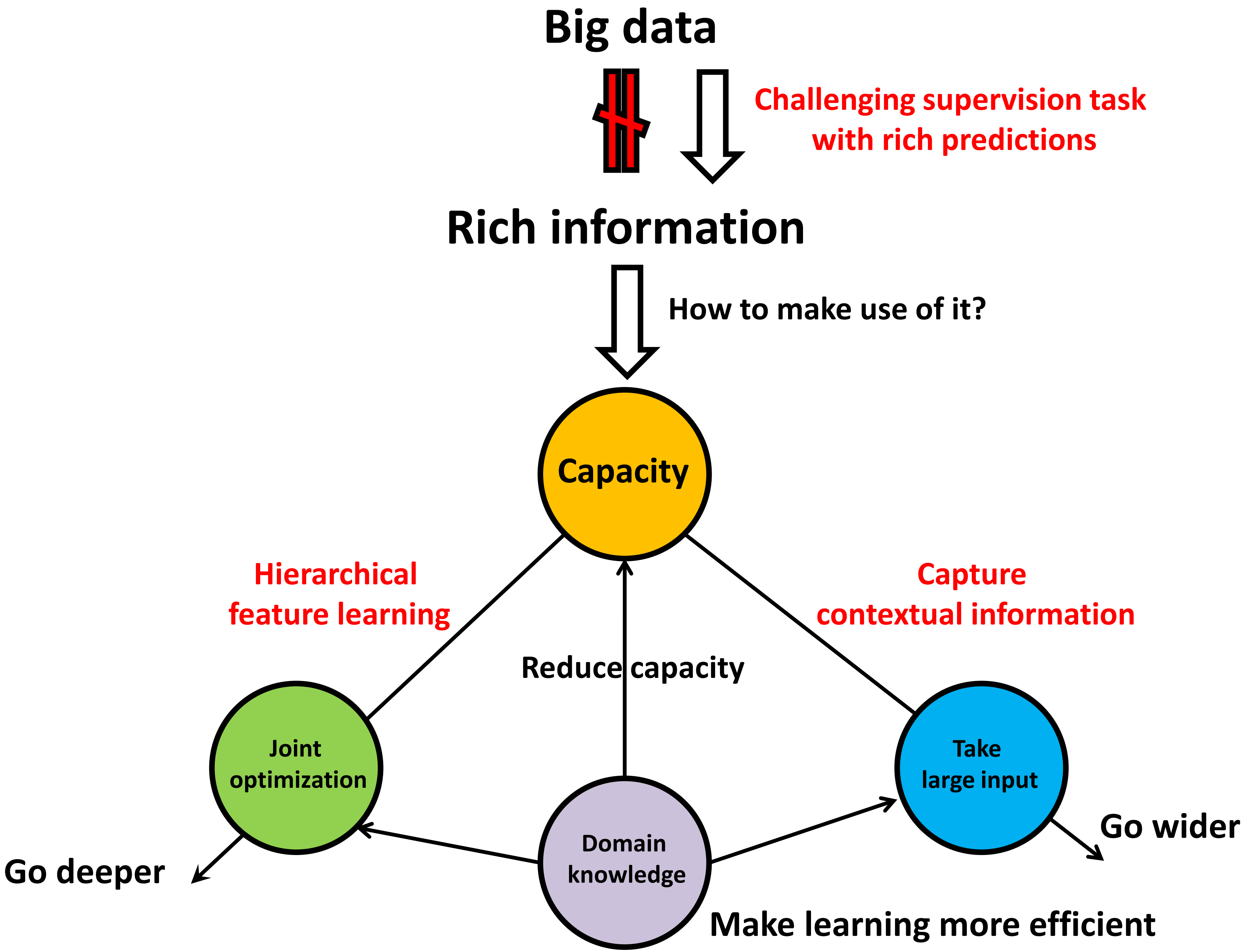
- P. Luo, X. Wang and X. Tang, "Hierarchical Face Parsing via Deep Learning," CVPR 2012

Motivations

- Recast face segmentation as a cross-modality data transformation problem
- Cross modality autoencoder
- Data of two different modalities share the same representations in the deep model
- Deep models can be used to learn shape priors for segmentation

Training Segmentators




Big data
Challenging supervision task
with rich predictions
Rich information
How to make use of it?
Capacity
Hierarchical
feature learning
Capture
contextual information
Reduce capacity
Joint
optimization
Take
large input
Go wider
Domain
knowledge
Go deeper
Make learning more efficient

Summary

- Automatically learns hierarchical feature representations from data and disentangles hidden factors of input data through multi-level nonlinear mappings
- For some tasks, the expressive power of deep models increases exponentially as their architectures go deep
- Jointly optimize all the components in a vision and crate synergy through close interactions among them
- Benefitting the large learning capacity of deep models, we also recast some classical computer vision challenges as high-dimensional data transform problems and solve them from new perspectives
- It is more effective to train deep models with challenging tasks and rich predictions

References

- D. E. Rumelhart, G. E. Hinton, R. J. Williams, “Learning Representations by Back-propagation Errors,” Nature, Vol. 323, pp. 533-536, 1986.
- N. Kruger, P. Janssen, S. Kalkan, M. Lappe, A. Leonardis, J. Piater, A. J. Rodriguez-Sanchez, L. Wiskott, “Deep Hierarchies in the Primate Visual Cortex: What Can We Learn For Computer Vision?” IEEE Trans. PAMI, Vol. 35, pp. 1847-1871, 2013.
- A. Krizhevsky, L. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” Proc. NIPS, 2012.
- Y. Sun, X. Wang, and X. Tang, “Deep Learning Face Representation by Joint Identification-Verification,” NIPS, 2014.
- K. Fukushima, “Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position,” Biological Cybernetics, Vol. 36, pp. 193-202, 1980.
- Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based Learning Applied to Document Recognition,” Proceedings of the IEEE, Vol. 86, pp. 2278-2324, 1998.
- G. E. Hinton, S. Osindero, and Y. Teh, “A Fast Learning Algorithm for Deep Belief Nets,” Neural Computation, Vol. 18, pp. 1527-1544, 2006.

- G. E. Hinton and R. R. Salakhutdinov, “Reducing the Dimensionality of Data with Neural Networks,” Science, Vol. 313, pp. 504-507, July 2006.
- Z. Zhu, P. Luo, X. Wang, and X. Tang, “Deep Learning Identity Face Space,” Proc. ICCV, 2013.
- Z. Zhu, P. Luo, X. Wang, and X. Tang, “Deep Learning and Disentangling Face Representation by Multi-View Perception,” NIPS 2014.
- Y. Sun, X. Wang, and X. Tang, “Deep Learning Face Representation from Predicting 10,000 classes,” Proc. CVPR, 2014.
- J. Hastad, “Almost Optimal Lower Bounds for Small Depth Circuits,” Proc. ACM Symposium on Theory of Computing, 1986.
- J. Hastad and M. Goldmann, “On the Power of Small-Depth Threshold Circuits,” Computational Complexity, Vol. 1, pp. 113-129, 1991.
- A. Yao, “Separating the Polynomial-time Hierarchy by Oracles,” Proc. IEEE Symposium on Foundations of Computer Science, 1985.
- Sermnet, K. Kavukcuoglu, S. Chintala, and LeCun, “Pedestrian Detection with Unsupervised Multi-Stage Feature Learning,” CVPR 2013.
- W. Ouyang and X. Wang, “Joint Deep Learning for Pedestrian Detection,” Proc. ICCV, 2013.
- P. Luo, X. Wang and X. Tang, “Hierarchical Face Parsing via Deep Learning,” Proc. CVPR, 2012.
- Honglak Lee, “Tutorial on Deep Learning and Applications,” NIPS 2010.

Outline

- Introduction to deep learning
- **Deep learning for object recognition**
- Deep learning for object segmentation
- Deep learning for object detection
- Open questions and future works

Part II: Deep Learning Object Recognition

- Deep learning for object recognition on ImageNet
- Deep learning for face recognition
  - Learn identity features from joint verification-identification signals
  - Learn 3D face models from 2D images

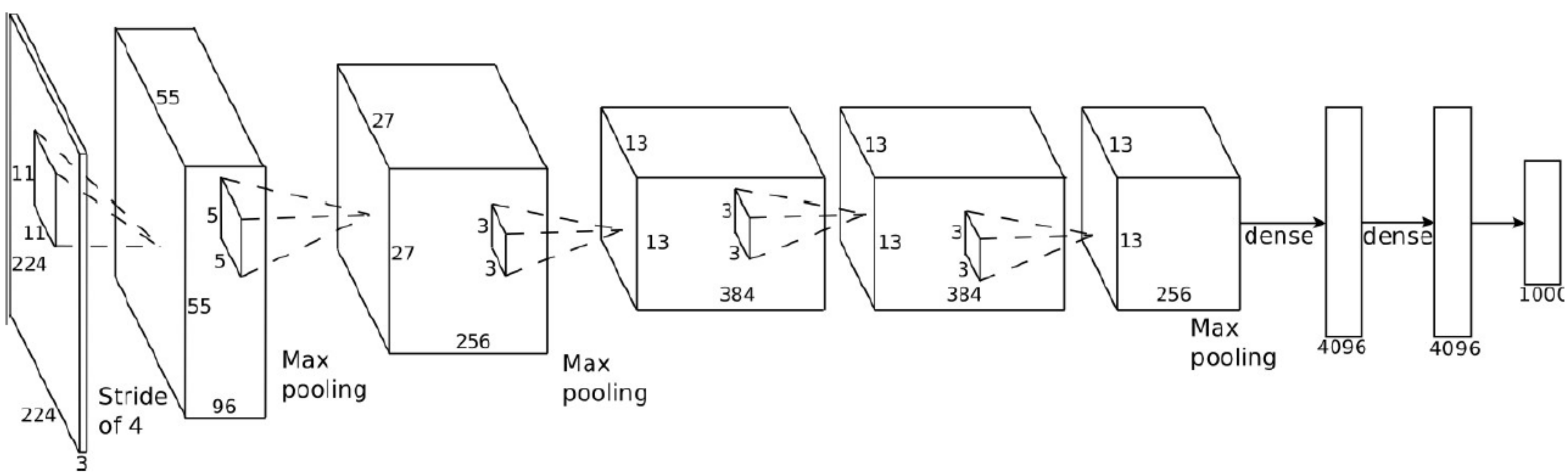
CNN for Object Recognition on ImageNet

- Krizhevsky, Sutskever, and Hinton, NIPS 2012
- Trained on one million images of 1000 categories collected from the web with two GPUs; 2GB RAM on each GPU; 5GB of system memory
- Training lasts for one week

Rank	Name	Error rate	Description
1	**U. Toronto**	0.15315	Deep learning
2	U. Tokyo	0.26172	Hand-crafted features and learning models. Bottleneck.
3	U. Oxford	0.26979	
4	Xerox/INRIA	0.27058	

Model Architecture

- Max-pooling layers follow 1st, 2nd, and 5th convolutional layers
- The number of neurons in each layer is given by 253440, 186624, 64896, 43264, 4096, 4096, 1000
- 650000 neurons, 60 million parameters, 630 million connections



Normalization

- Normalize the input by subtracting the mean image on the training set

Input image (256 x 256)

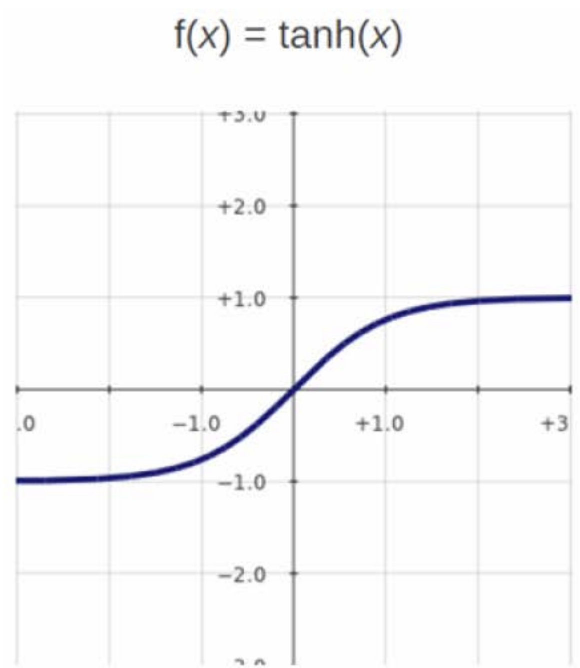
−

Mean image

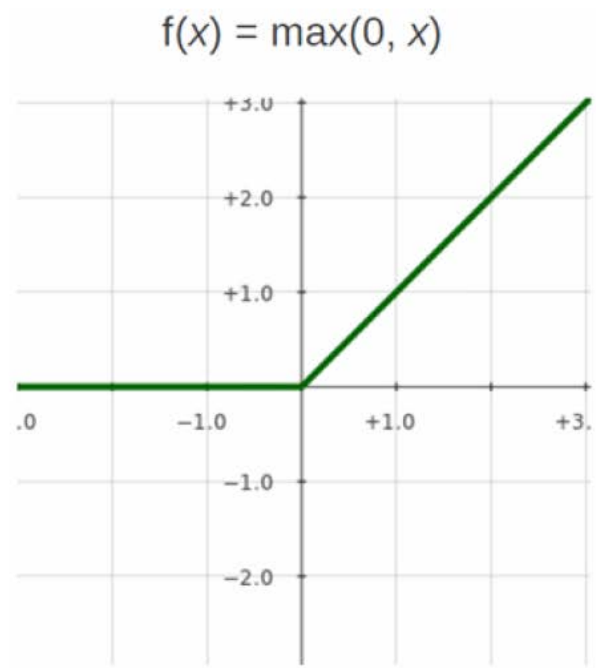
Krizhevsky 2012

Activation Function

- Rectified linear unit leads to sparse responses of neurons, such that weights can be effectively updated with BP



Sigmoid (slow to train)

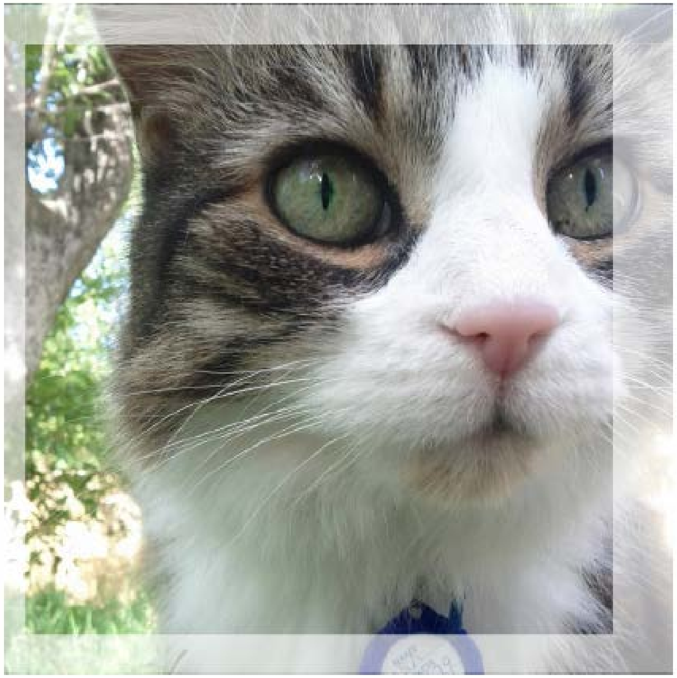


Rectified linear unit (quick to train) √

Krizhevsky 2012

Data Augmentation

- The neural net has 60M parameters and it overfits
- Image regions are randomly cropped with shift; their horizontal reflections are also included

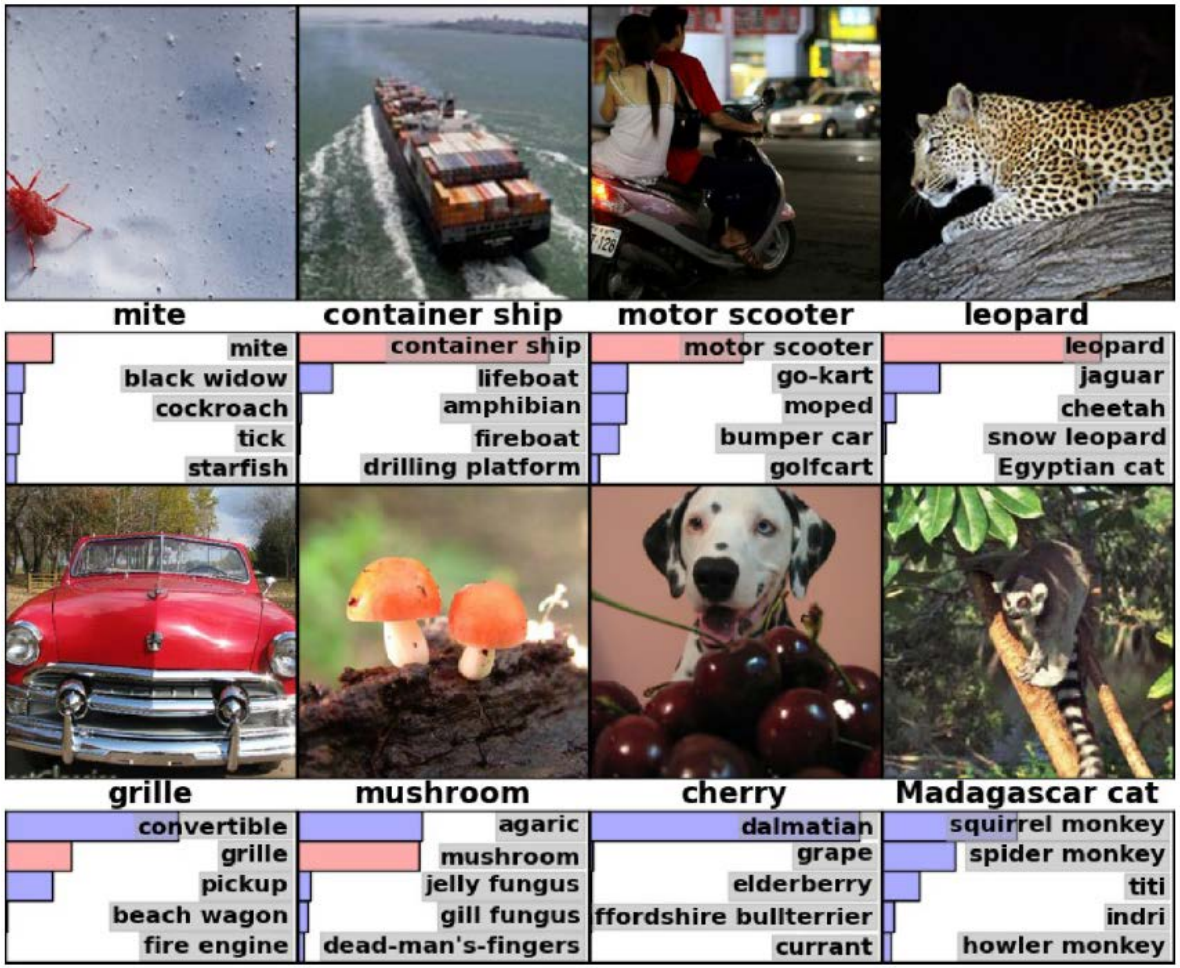


Krizhevsky 2012

Dropout

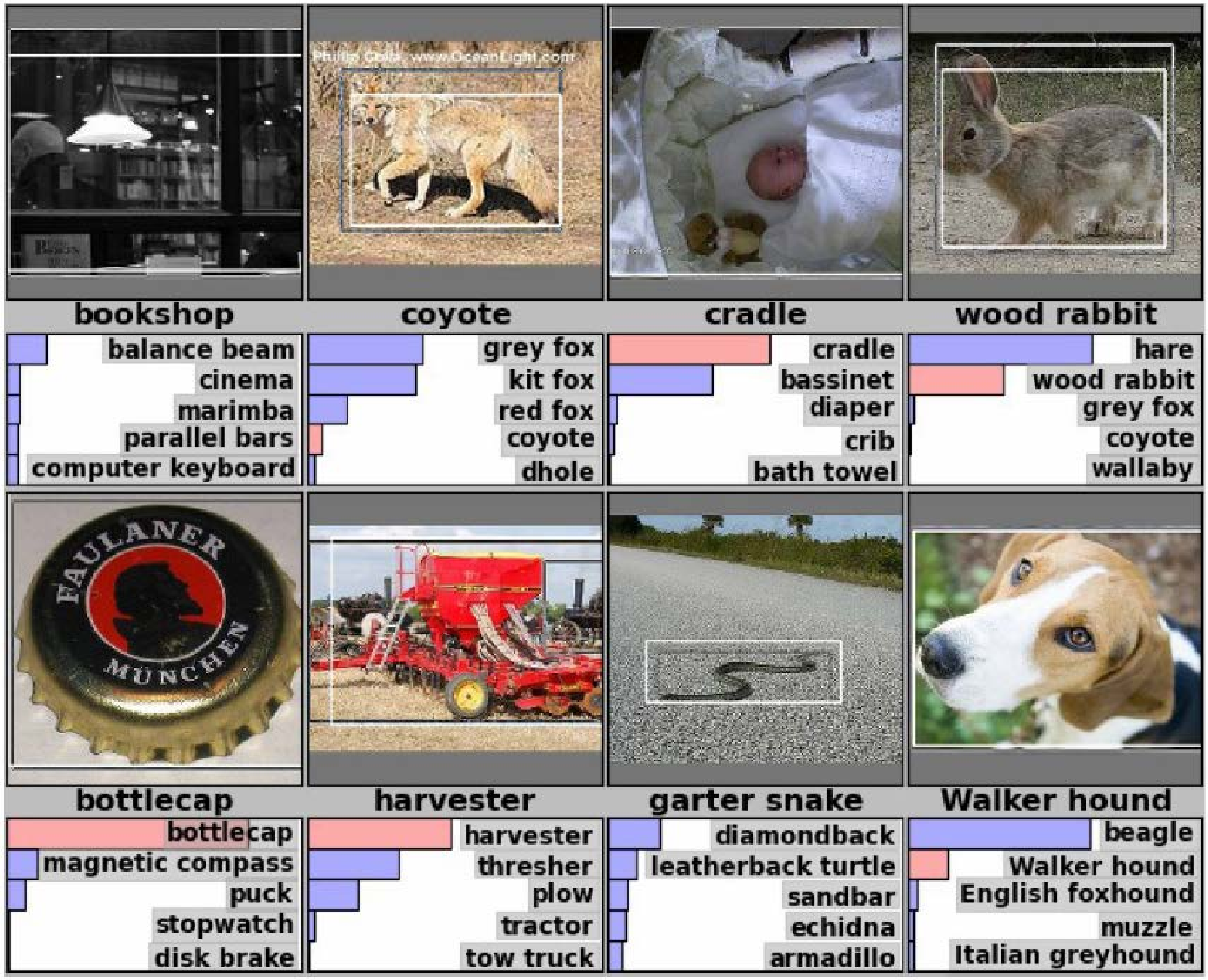
- Randomly set some input features and the outputs of hidden units as zero during the training process
- Feature co-adaptation: a feature is only helpful when other specific features are present
  - Because of the existence of noise and data corruption, some features or the responses of hidden nodes can be misdetected
- Dropout prevents feature co-adaptation and can significantly improve the generalization of the trained network
- Can be considered as another approach to regularization
- It can be viewed as averaging over many neural networks
- Slower convergence

Classification Result



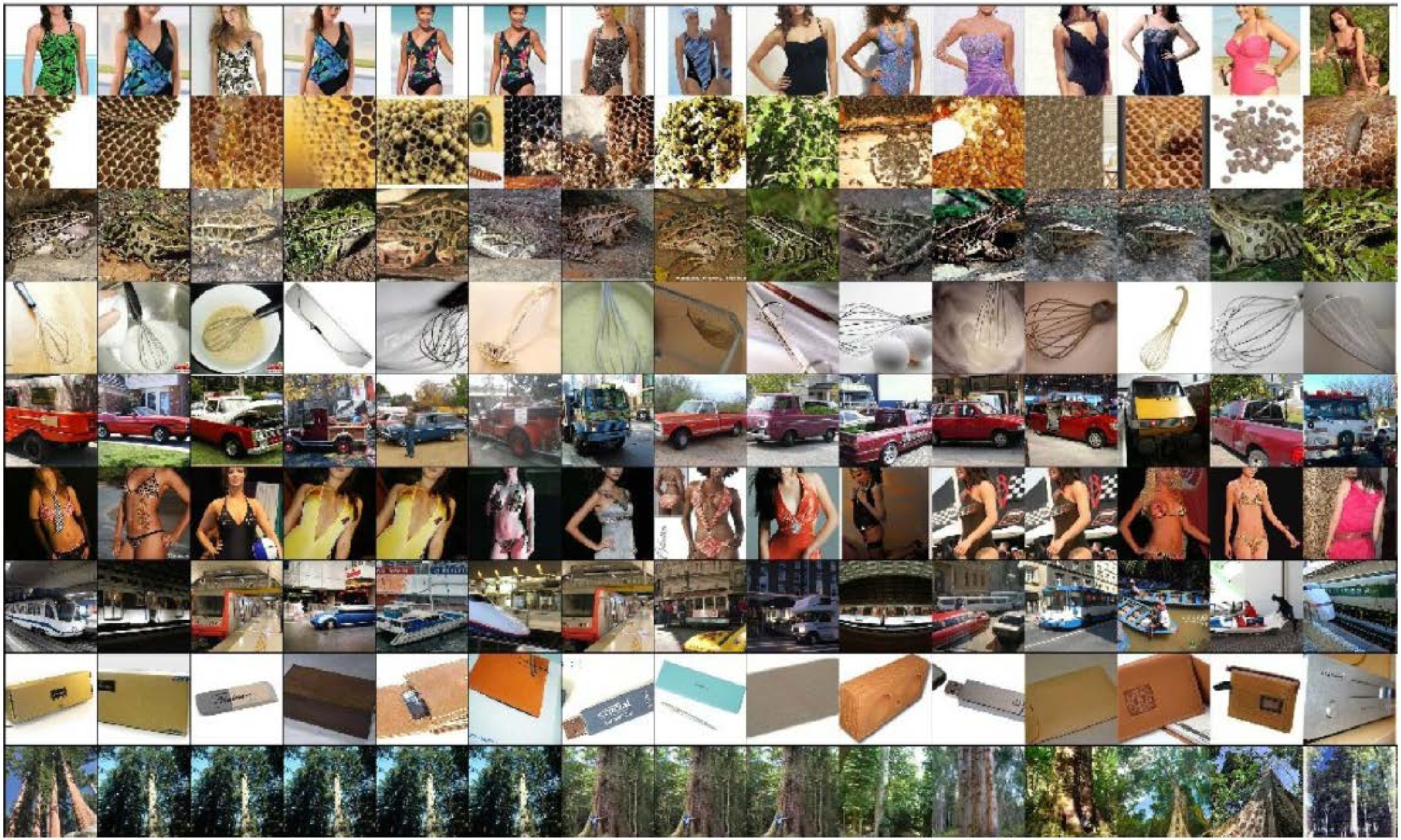
Krizhevsky 2012

Detection Result



Krizhevsky 2012

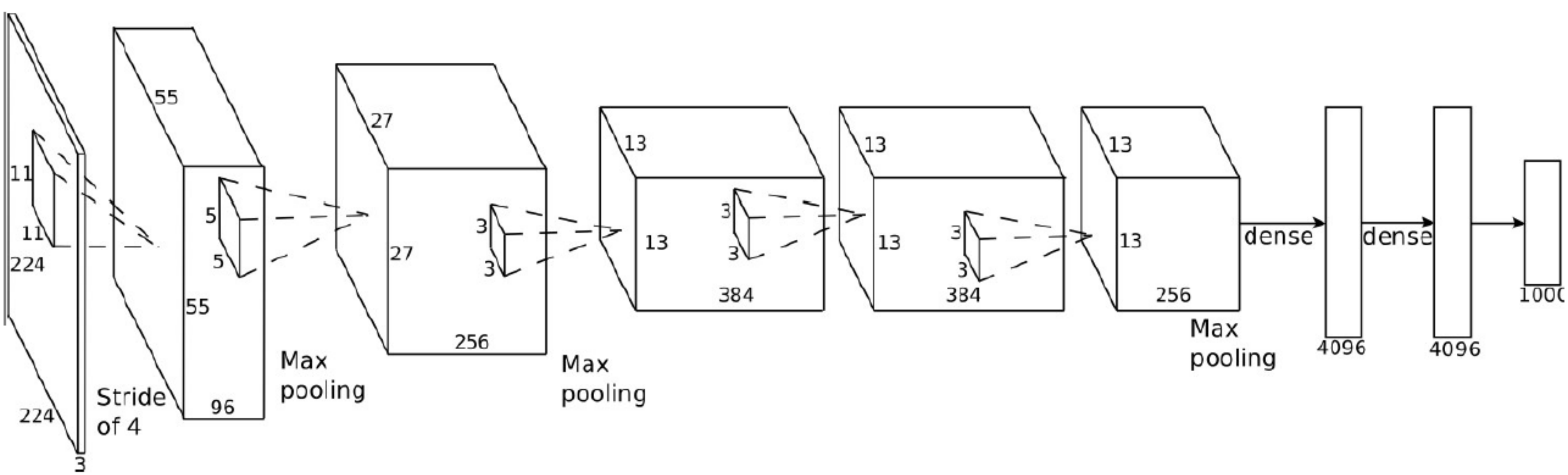
Image Retrieval



Krizhevsky 2012

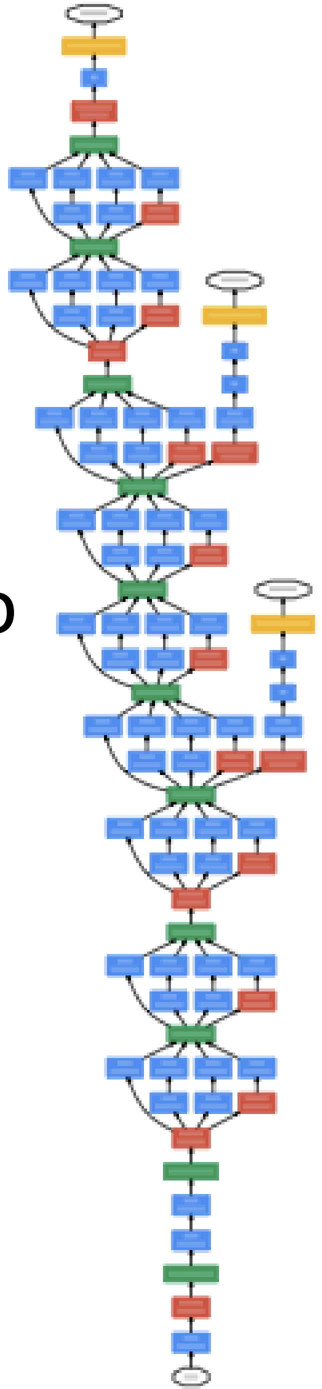
Adaptation to Smaller Datasets

- Directly use the feature representations learned from ImageNet and replace handcrafted features with them in image classification, scene recognition, fine grained object recognition, attribute recognition, image retrieval (Razavian et al. 2014, Gong et al. 2014)
- Use ImageNet to pre-train the model (good initialization), and use target dataset to fine-tune it (Girshick et al. CVPR 2014)
- Fix the bottom layers and only fine tune the top layers



GoogLeNet

- More than 20 layers
- Add supervision at multiple layers
- The error rate is reduced from 15.3% to 6.6%



Deep Learning Object Recognition

- Deep learning for object recognition on ImageNet
- **Deep learning for face recognition**
  - **Learn identity features from joint verification-identification signals**
  - Learn 3D face models from 2D images

Deep Learning Results on LFW

Method	Accuracy (%)	# points	# training images
Huang et al. CVPR'12	87%	3	Unsupervised
Sun et al. ICCV'13	92.52%	5	87,628
DeepFace (CVPR'14)	97.35%	6 + 67	7,000,000
Sun et al. (CVPR'14)	97.45%	5	202,599
Sun et al. (NIPS'14)	99.15%	18	202,599
New: DeepID2+ (CVPR'15)	**99.47%**	**18**	**450,000**

- The first deep learning work on face recognition was done by Huang et al. in 2012. With unsupervised learning, the accuracy was 87%
- Our work at ICCV'13 achieved result (92.52%) comparable with state-of-the-art
- Our work at CVPR'14 reached **97.45%** close to "human cropped" performance (**97.53%**)
- DeepFace developed by Facebook also at CVPR'14 used 73-point 3D face alignment and 7 million training data (35 times larger than us)
- Our most recent work reached **99.15%** close to "human funneled" performance (**99.20%**)

Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. CVPR, 2015.

Closed- and open-set face identification on LFW

Method	Rank-1 (%)	DIR @ 1% FAR (%)
COST-S1 [1]	56.7	25
COST-S1+s2 [1]	66.5	35
DeepFace [2]	64.9	44.5
DeepFace+ [3]	82.5	61.9
DeepID2 [4]	**91.1**	**61.6**
DeepID2+ [5]	**95.0**	**80.7**

[1] L. Best-Rowden, H. Han, C. Otto, B. Klare, and A. K. Jain. Unconstrained face recognition: Identifying a person of interest from a media collection. *TR MSU-CSE-14-1,* 2014.

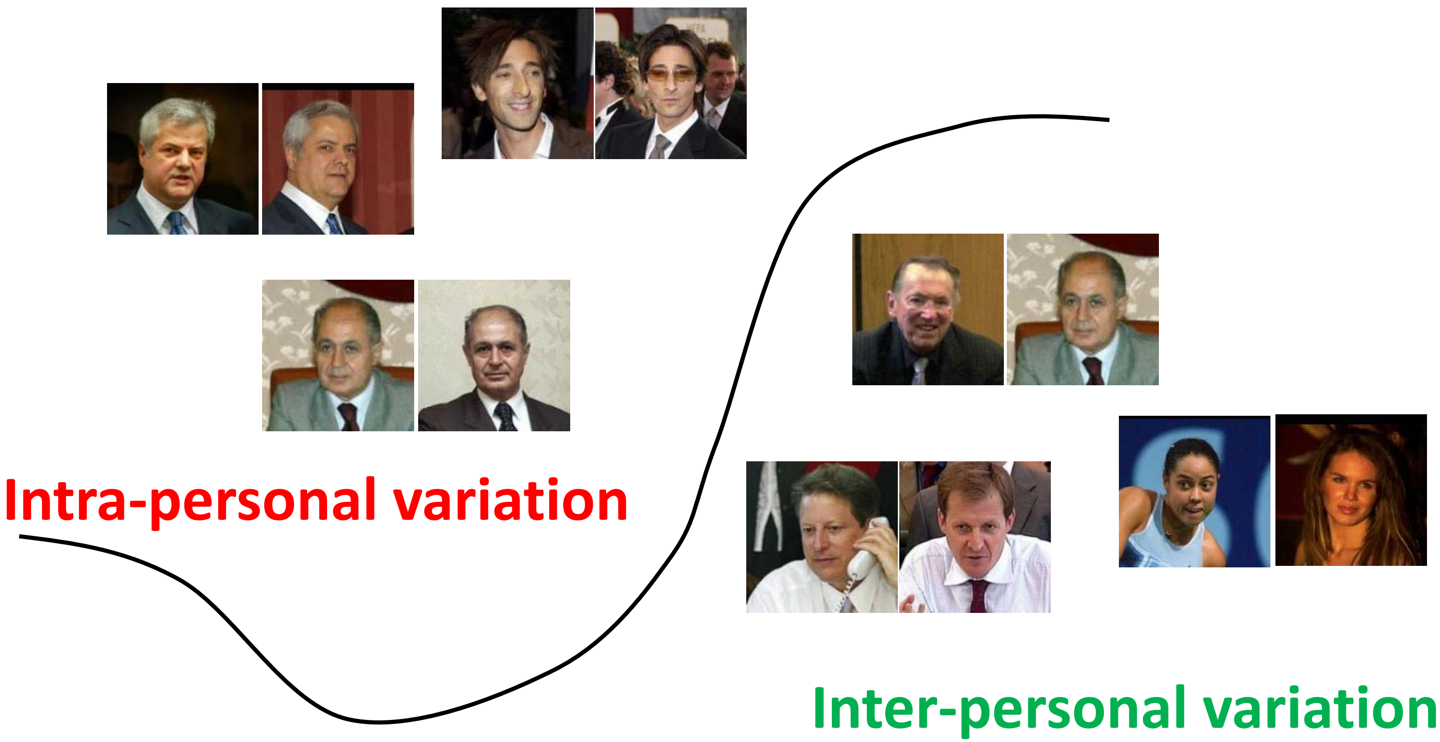
[2] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. DeepFace: Closing the gap to human-level performance in face verifica- tion. In *Proc. CVPR,* 2014.

[3] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Web- scale training for face identification. Technical report, arXiv:1406.5266, 2014.

[4] Y. Sun, X. Wang, and X. Tang. Deep Learning Face Representation by Joint Identification-Verification. NIPS, 2014.

[5] Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. CVPR, 2015.

Eternal Topic on Face Recognition



How to separate the two types of variations?

Are they the same person or not?


Nicole Kidman


Nicole Kidman

Are they the same person or not?

Coo d’Este

Melina Kanakaredes

Are they the same person or not?

Elijah Wood　　　　Stefano Gabbana

Are they the same person or not?

Jim O’Brien

Jim O’Brien

Are they the same person or not?

Jacquline Obradors Julie Taymor

- Out of 6000 image pairs on the LFW test set, 51 pairs are misclassified with the deep model
- We randomly mixed them and presented them to 10 Chinese subjects for evaluation. Their averaged verification accuracy is 56%, close to random guess (50%)

Linear Discriminate Analysis

$$\mathbf{W}^* = \arg\max_{\mathbf{W}} \frac{|\mathbf{W}^t \mathbf{S}_b \mathbf{W}|}{|\mathbf{W}^t \mathbf{S}_w \mathbf{W}|}$$

$$\mathbf{S}_b = \sum n_k (\bar{\mathbf{x}}_k - \bar{x})(\bar{\mathbf{x}}_k - \bar{x})^t \propto \sum (\bar{\mathbf{x}}_k - \bar{\mathbf{x}}_{k'})(\bar{\mathbf{x}}_k - \bar{\mathbf{x}}_{k'})^t$$

$$\mathbf{S}_w = \sum_k \sum_{i \in C_k} (\mathbf{x}_i - \bar{\mathbf{x}}_k)(\mathbf{x}_i - \bar{\mathbf{x}}_k)^t \propto \sum_{(i,j) \in \Omega} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^t$$

$$\mathbf{W}^* = \arg\max_{\mathbf{W}} |\mathbf{W}^t \mathbf{S}_b \mathbf{W}| \quad s.t. \quad |\mathbf{W}^t \mathbf{S}_w \mathbf{W}| = 1$$

LDA seeks for linear feature mapping which maximizes the distance between class centers under the constraint what the intrapersonal variation is constant

$$\mathbf{y}_i = f(\mathbf{x}_i) = \mathbf{W}^t \mathbf{x}_i$$

$$f^* = \arg\max_{f^*} \sum_{k,k'} |f(\bar{\mathbf{x}}_k) - f(\bar{\mathbf{x}}_{k'})|^2$$

$$s.t. \sum_{(i,j)\in\Omega_I} |f(\mathbf{x}_i) - f(\mathbf{x}_j)|^2 = 1$$

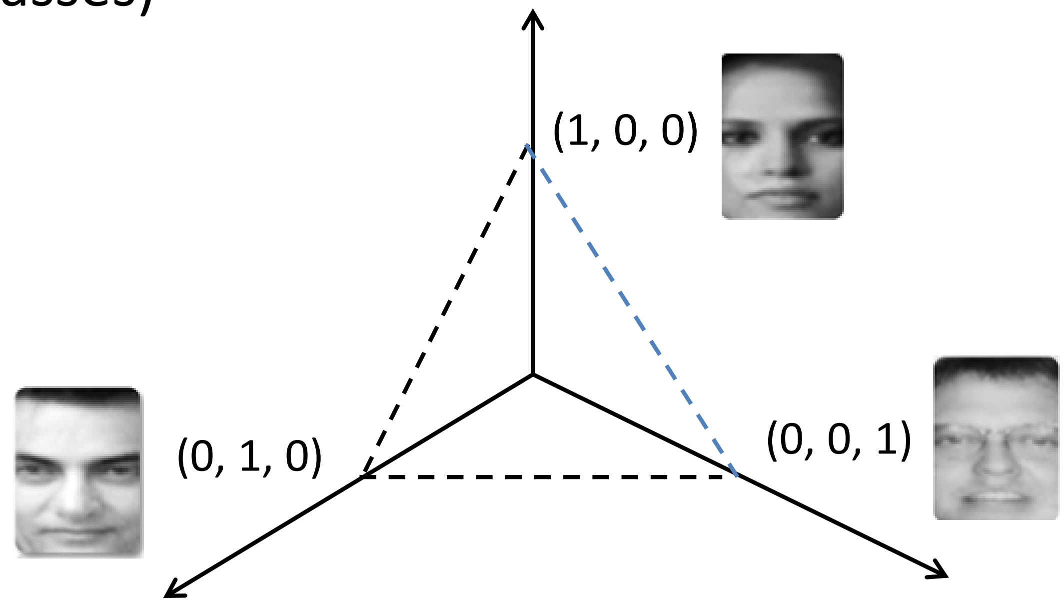
Deep Learning for Face Recognition

- Extract identity preserving features through hierarchical nonlinear mappings
- Model complex intra- and inter-personal variations with large learning capacity

Learn Identity Features from Different Supervisory Tasks

- Face identification: classify an image into one of N identity classes
  - multi-class classification problem
- Face verification: verify whether a pair of images belong to the same identity or not
  - binary classification problem

Minimize the intra-personal variation under the constraint that the distance between classes is constant (i.e. contracting the volume of the image space without reducing the distance between classes)



$$\mathbf{y} = f(\mathbf{x}); \quad g = \text{softmax}()$$

$$f^* = \arg\min_f \sum_{(i,j)\in\Omega_I} ||f(\mathbf{x}_i) - f(\mathbf{x}_j)||^2$$

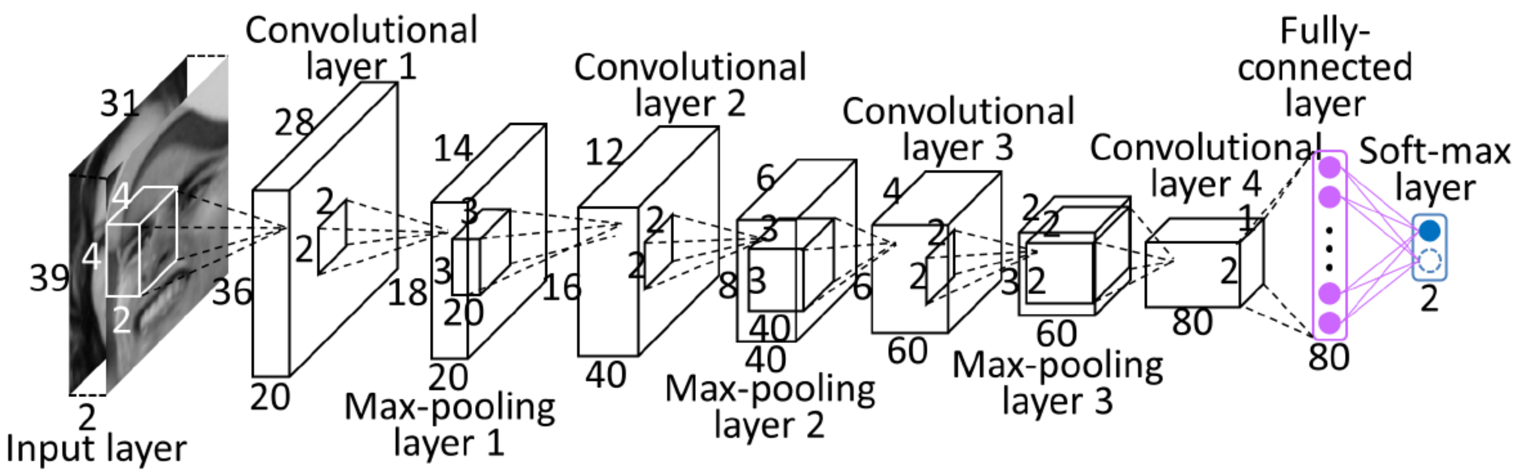
$$s.t. \;\; |g(f(\mathbf{x}_i)) - g(f(\mathbf{x}_j))| = 1, \;\; label(\mathbf{x}_i) \neq label(\mathbf{x}_j)$$

Learn Identity Features with Verification Signal

- Extract relational features with learned filter pairs

$$y^j = f\left(b^j + k^{1j} * x^1 + k^{2j} * x^2\right)$$

- These relational features are further processed through multiple layers to extract global features
- The fully connected layer can be used as features to combine with multiple ConvNets

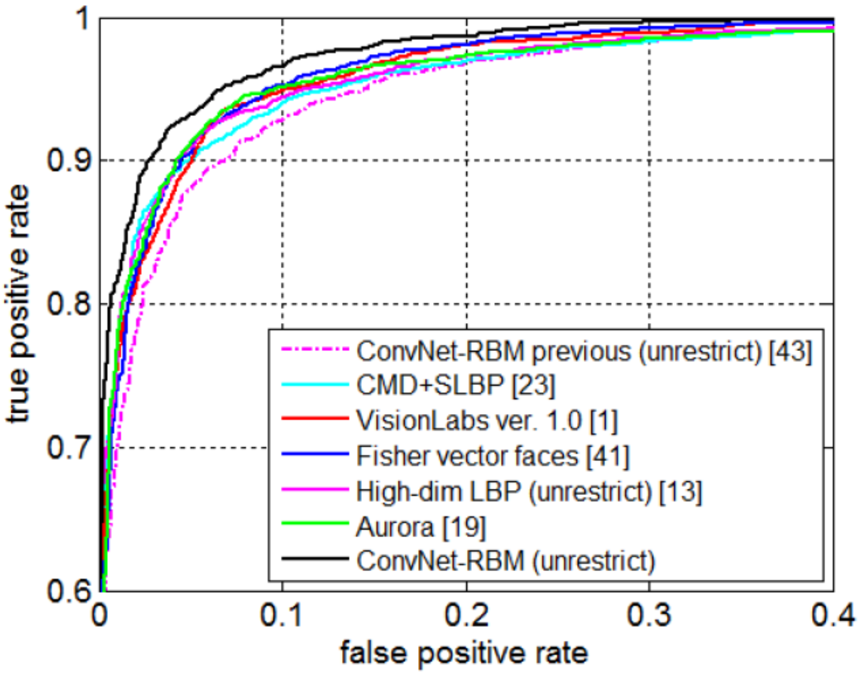


Y. Sun, X. Wang, and X. Tang, “Hybrid Deep Learning for Computing Face Similarities,” Proc. ICCV, 2013.

Results on LFW

- Unrestricted protocol without outside training data

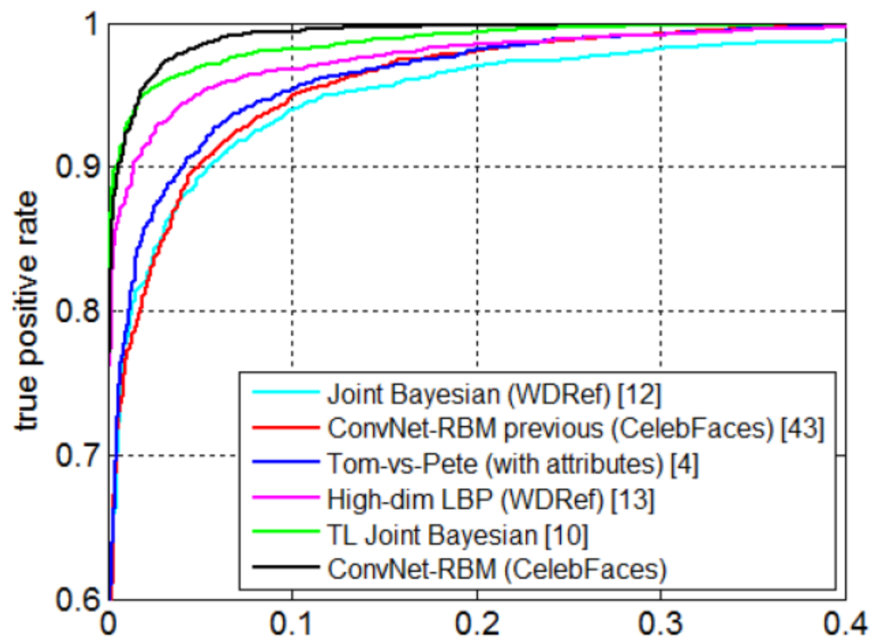
Method	Accuracy (%)
ConvNet-RBM previous [43]	91.75 ± 0.48
VMRS [3]	92.05 ± 0.45
CMD+SLBP [23]	92.58 ± 1.36
VisionLabs ver. 1.0 [1]	92.90 ± 0.31
Fisher vector faces [41]	93.03 ± 1.05
High-dim LBP [13]	93.18 ± 1.07
Aurora [19]	93.24 ± 0.44
ConvNet-RBM	$\mathbf{93.83 \pm 0.52}$



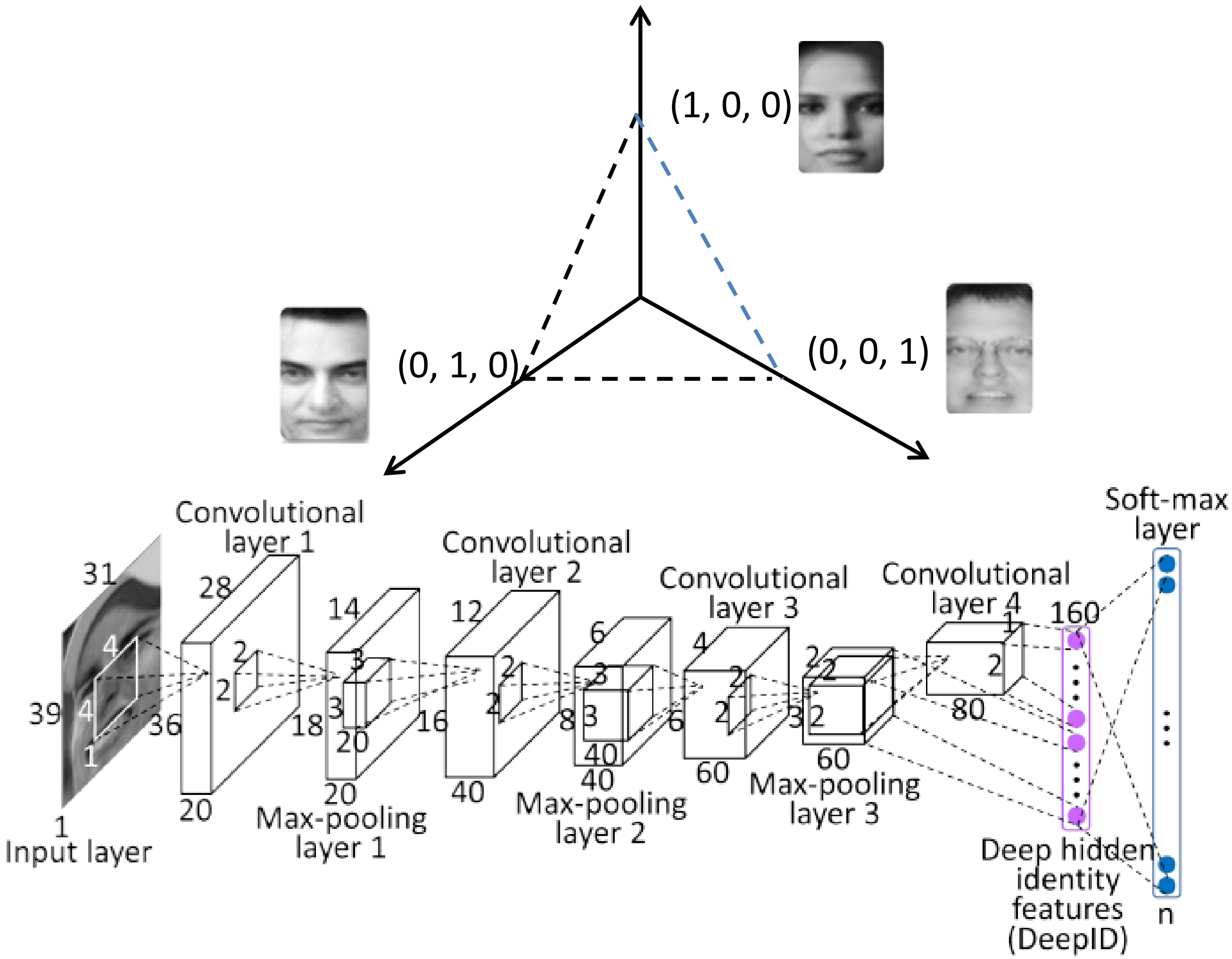
Results on LFW

- Unrestricted protocol using outside training data

Method	Accuracy (%)
Joint Bayesian [12]	92.42 ± 1.08
ConvNet-RBM previous [43]	92.52 ± 0.38
Tom-vs-Pete (with attributes) [4]	93.30 ± 1.28
High-dim LBP [13]	95.17 ± 1.13
TL Joint Bayesian [10]	96.33 ± 1.08
ConvNet-RBM	$\mathbf{97.08 \pm 0.28}$

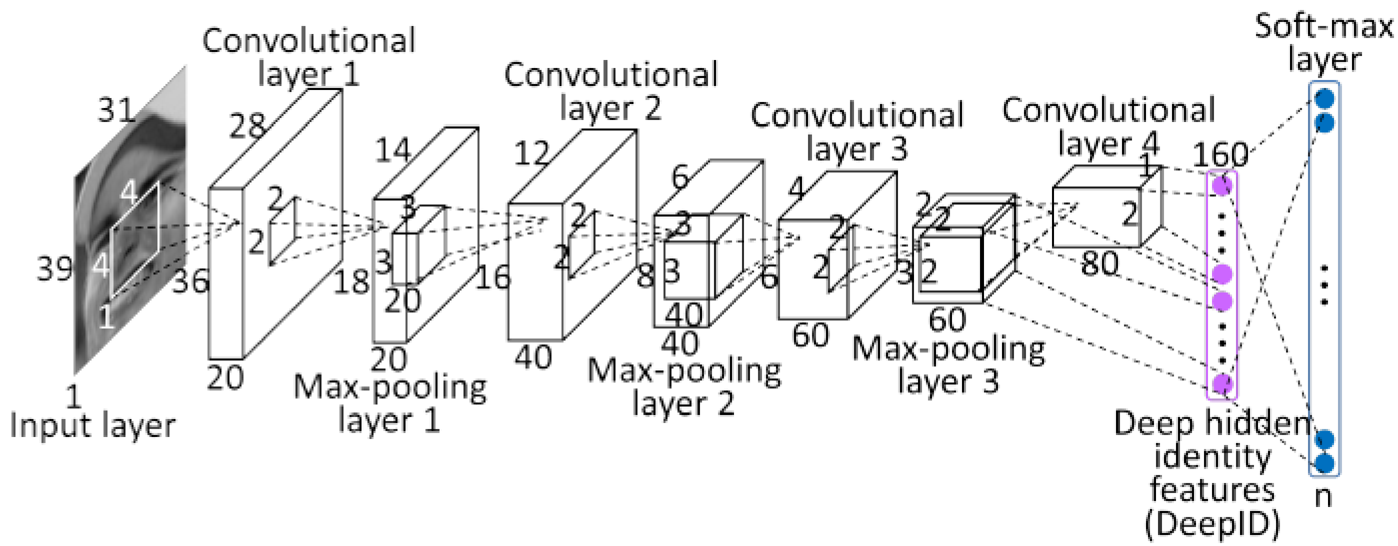


DeepID: Learn Identity Features with Identification Signal

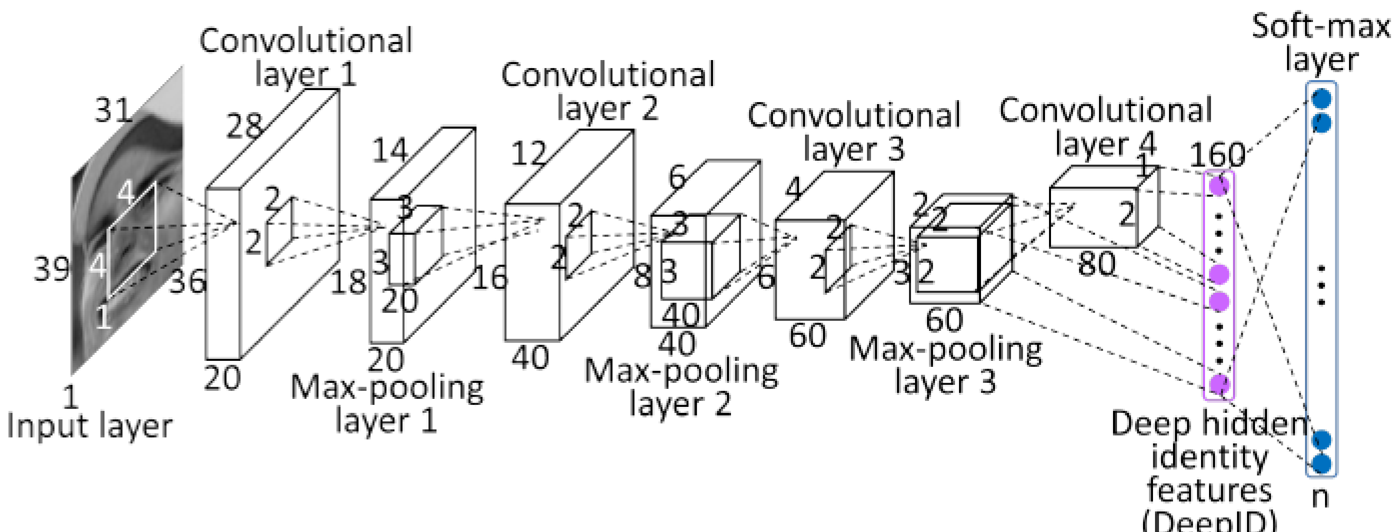


Y. Sun, X. Wang, and X. Tang, "Deep Learning Face Representation from Predicting 10,000 classes," Proc. CVPR, 2014.

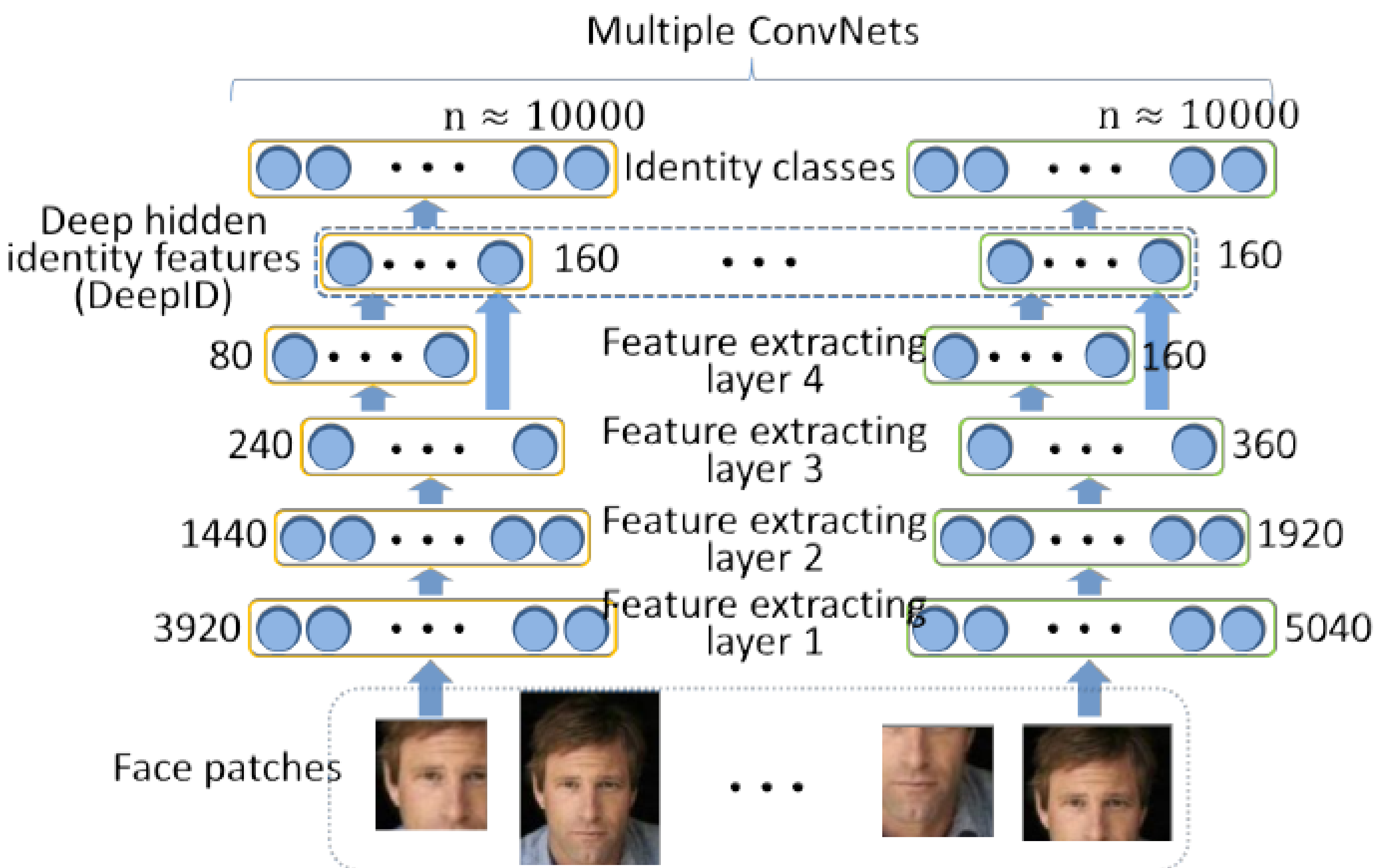
- During training, each image is classified into 10,000 identities with 160 identity features in the top layer
- These features keep rich inter-personal variations
- Features from the last two convolutional layers are effective
- The hidden identity features can be well generalized to other tasks (e.g. verification) and identities outside the training set



- High-dimensional prediction is more challenging, but also adds stronger supervision to the network
- As adding the number of classes to be predicted, the generalization power of the learned features also improves



Extract Features from Multiple ConvNets



Learn Identity Features with Identification Signal

- After combining hidden identity features from multiple CovNets and further reducing dimensionality with PCA, each face image has 150-dimenional features as signature
- These features can be further processed by other classifiers in face verification. Interestingly, we find Joint Bayesian is more effective than cascading another neural network to classify these features

DeepID2: Joint Identification-Verification Signals

- Every two feature vectors extracted from the same identity should are close to each other

$$\mathrm{Verif}(f_i, f_j, y_{ij}, \theta_{ve}) = \begin{cases} \frac{1}{2}\left\|f_i - f_j\right\|_2^2 & \text{if } y_{ij} = 1 \\ \frac{1}{2}\max\left(0, m - \left\|f_i - f_j\right\|_2\right)^2 & \text{if } y_{ij} = -1 \end{cases}$$

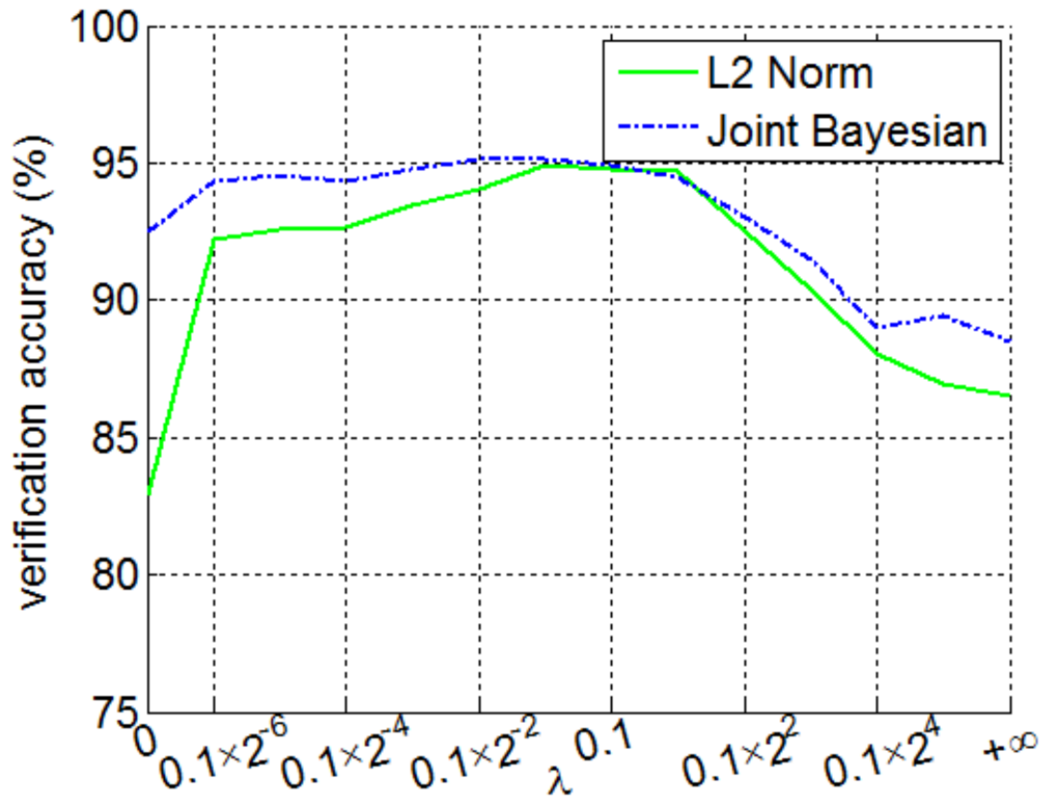
f_i and f_j are feature vectors extracted from two face images in comparison

y_{ij} = 1 means they are from the same identity; y_{ij} = -1means different identities

m is a margin to be learned

Y. Sun, X. Wang, and X. Tang. Deep Learning Face Representation by Joint Identification-Verification. NIPS, 2014.

Balancing Identification and Verification Signals with Parameter λ

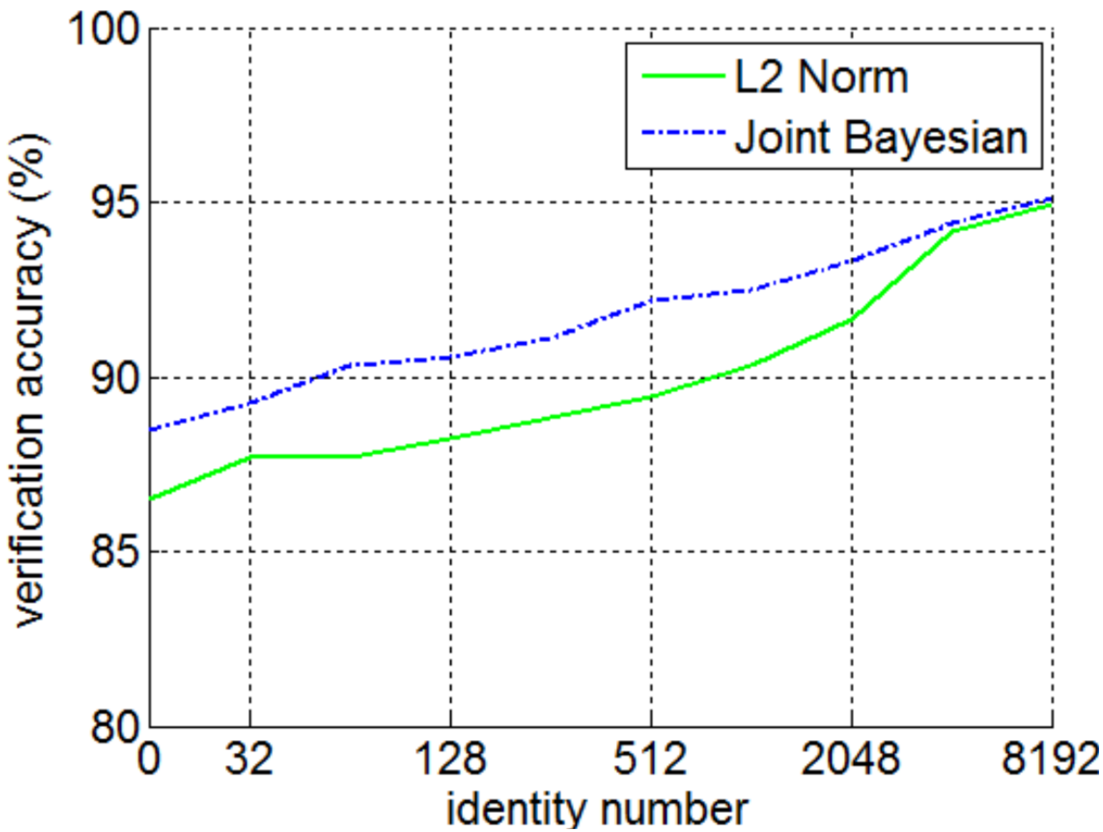


λ = 0: only identification signal

λ = +∞: only verification signal

Rich Identity Information Improves Feature Learning

- Face verification accuracies with the number of training identities

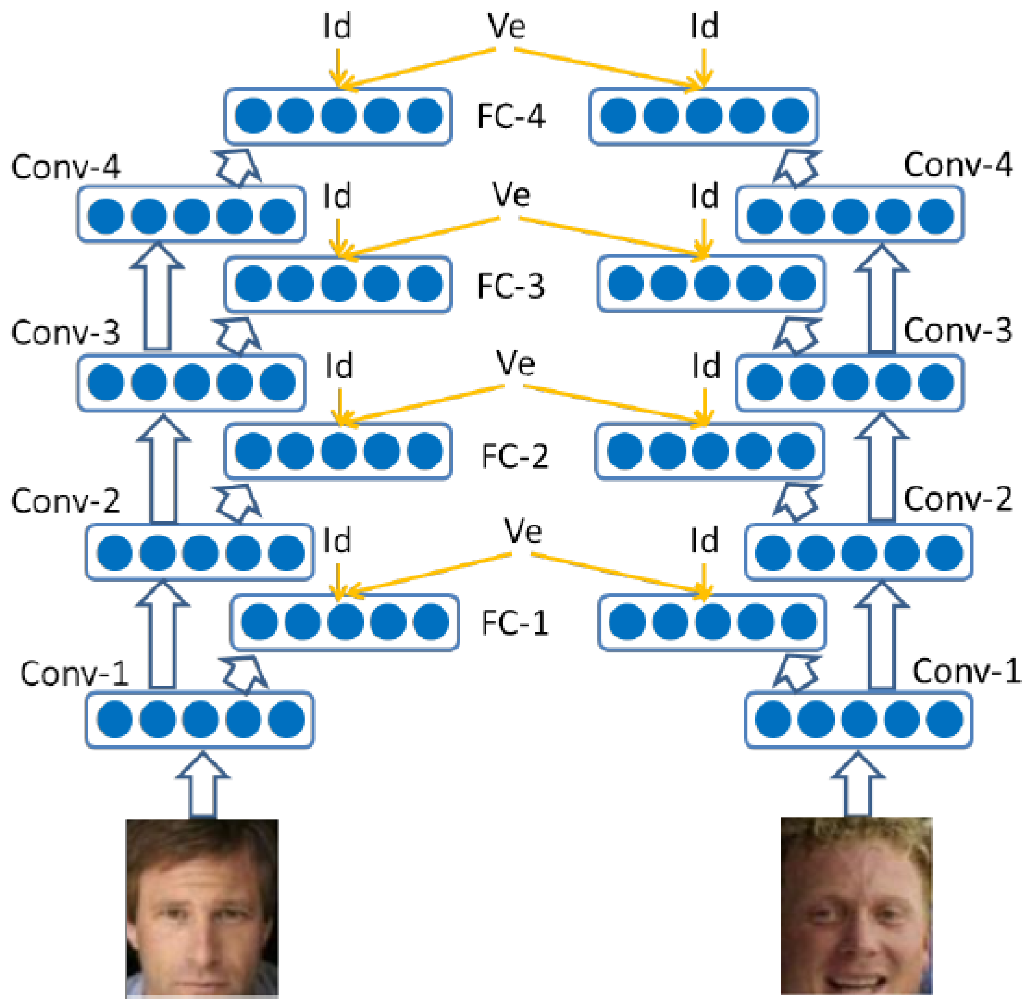


Summary of DeepID2

- 25 face regions at different scales and locations around landmarks are selected to build 25 neural networks
- All the 160 X 25 hidden identity features are further compressed into a 180-dimensional feature vector with PCA as a signature for each image
- With a single Titan GPU, the feature extraction process takes 35ms per image

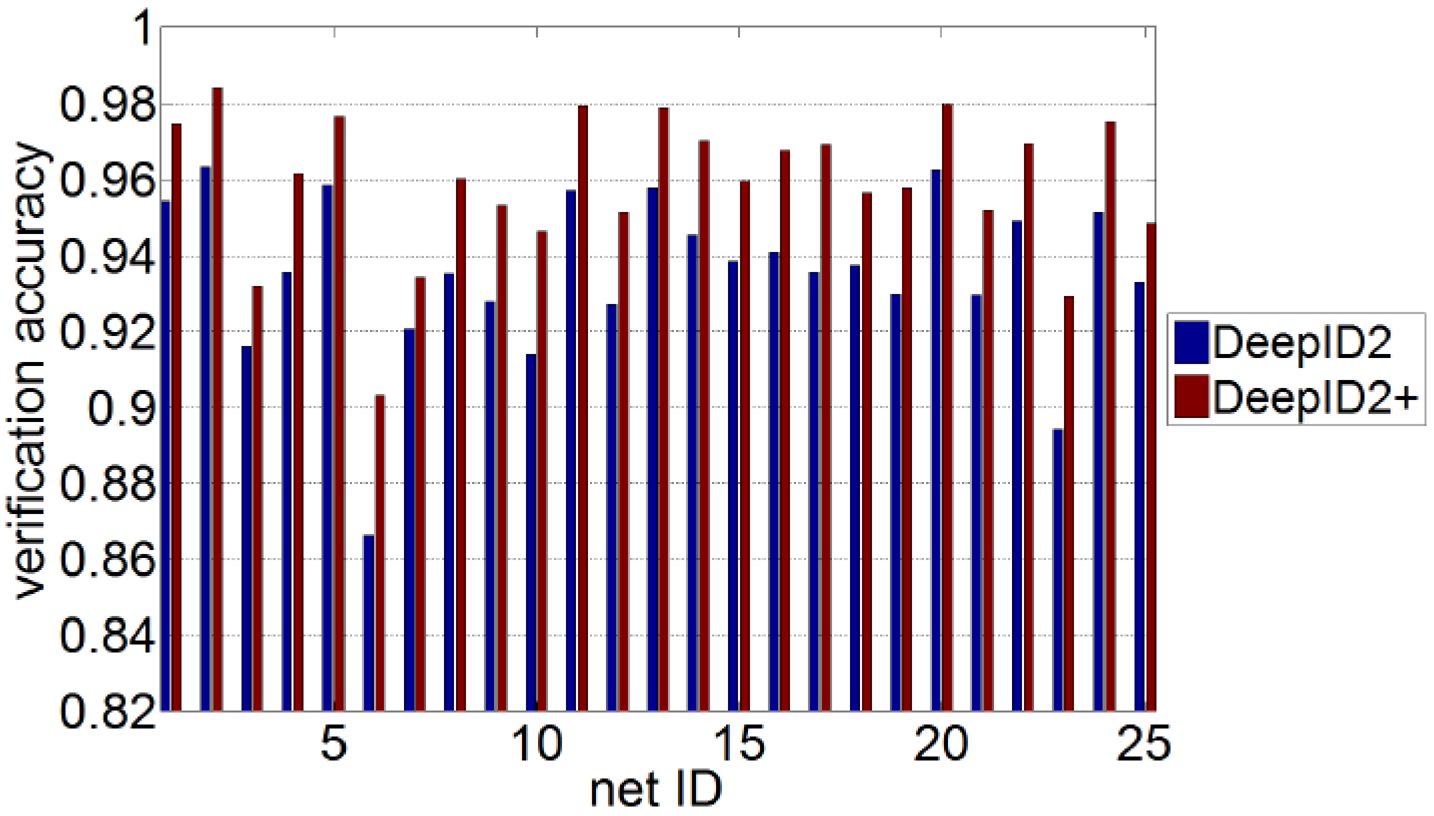
DeepID2+

- Larger net work structures
- Larger training data
- Adding supervisory signals at every layer



Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. CVPR, 2015.

Compare DeepID2 and DeepID2+ on LFW



Comparison of face verification accuracies on LFW with ConvNets trained on 25 face regions given in DeepID2

Best single model is improved from 96.72% to 98.70%

Final Result on LFW

Methods	High-dim LBP [1]	TL Joint Bayesian [2]	DeepFace [3]	DeepID [4]	DeepID2 [5]	DeepID2+ [6]
Accuracy (%)	95.17	96.33	97.35	97.45	99.15	99.47

[1] Chen, Cao, Wen, and Sun. Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification. CVPR, 2013.

[2] Cao, Wipf, Wen, Duan, and Sun. A practical transfer learning algorithm for face verification. ICCV, 2013.

[3] Taigman, Yang, Ranzato, and Wolf. DeepFace: Closing the gap to human-level performance in face verification. *CVPR,* 2014.

[4] Sun, Wang, and Tang. Deep learning face representation from predicting 10,000 classes. *CVPR,* 2014.

[5] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep Learning Face Representation by Joint Identification-Verification. NIPS, 2014.

[6] Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. CVPR, 2015.

Closed- and open-set face identification on LFW

Method	Rank-1 (%)	DIR @ 1% FAR (%)
COST-S1 [1]	56.7	25
COST-S1+s2 [1]	66.5	35
DeepFace [2]	64.9	44.5
DeepFace+ [3]	82.5	61.9
DeepID2	91.1	61.6
DeepID2+	**95.0**	**80.7**

[1] L. Best-Rowden, H. Han, C. Otto, B. Klare, and A. K. Jain. Unconstrained face recognition: Identifying a person of interest from a media collection. *TR MSU-CSE-14-1,* 2014.

[2] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. DeepFace: Closing the gap to human-level performance in face verifica- tion. In *Proc. CVPR,* 2014.

[3] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Web- scale training for face identification. Technical report, arXiv:1406.5266, 2014.

Face Verification on YouTube Faces

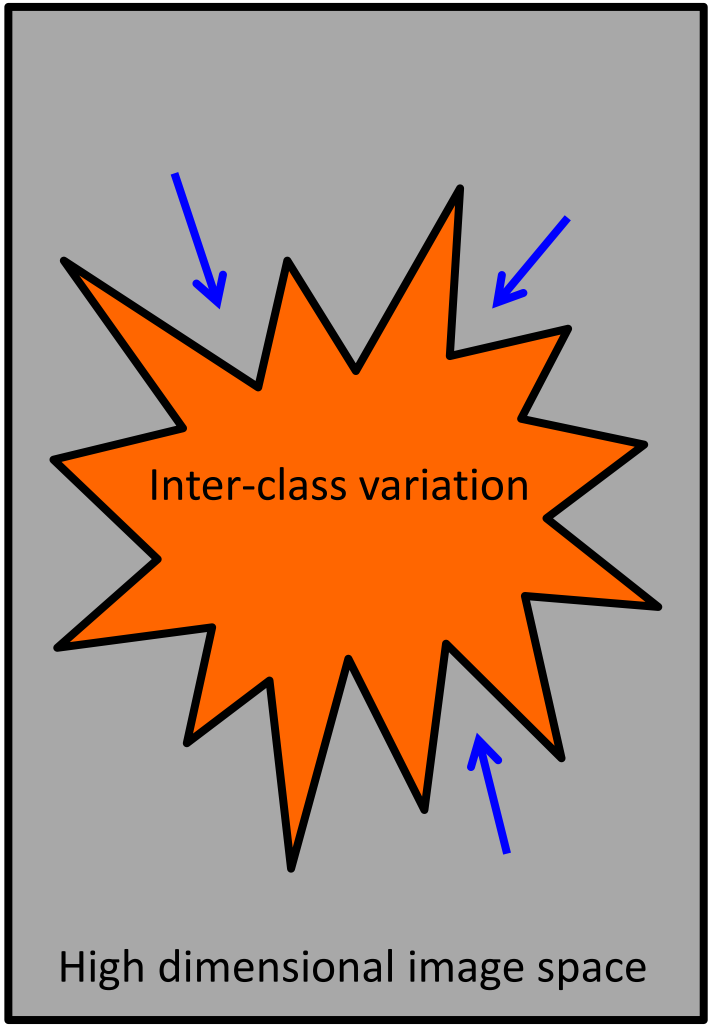
Methods	Accuracy (%)
LM3L [1]	81.3 ± 1.2
DDML (LBP) [2]	81.3 ± 1.6
DDML (combined) [2]	82.3 ± 1.5
EigenPEP [3]	84.8 ± 1.4
DeepFace [4]	91.4 ± 1.1
DeepID2+	93.2 ± 0.2

[1] J. Hu, J. Lu, J. Yuan, and Y. P. Tan, "Large margin multi-metric learning for face and kinship verification in the wild," ACCV 2014

[2] J. Hu, J. Lu, and Y. P. Tan, "Discriminative deep metric learning for face verification in the wild," CVPR 2014

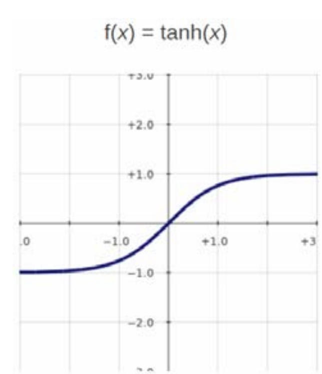
[3] H. Li, G. Hua, X. Shen, Z. Lin, and J. Brandt, "Eigen-pep for video face recognition," ACCV 2014

[4] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," CVPR 2014.

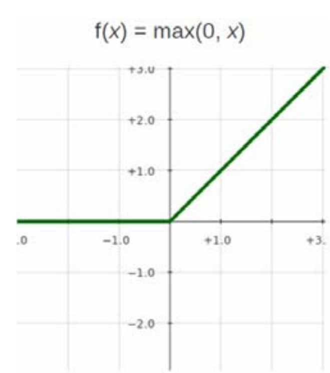


- **Linear transform**
- **Pooling**
- **Nonlinear mapping**

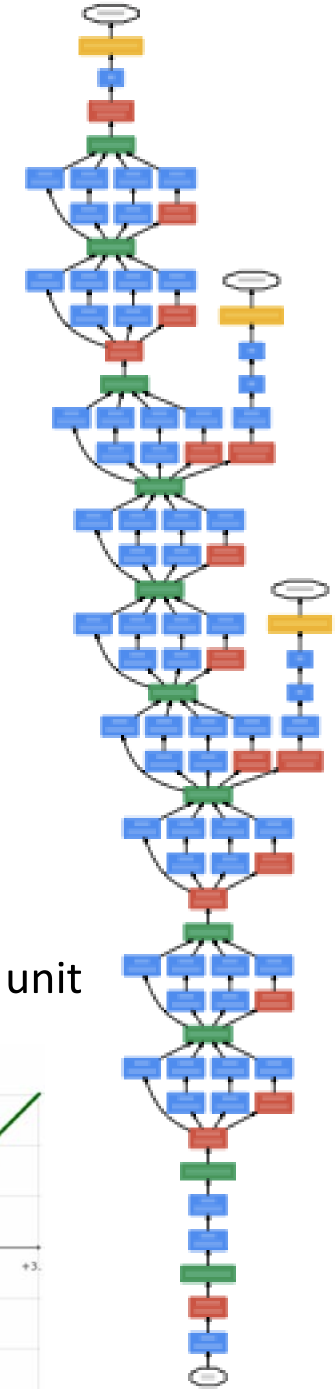
Sigmoid



Rectified linear unit



GoogLeNet



Unified subspace analysis

- Identification signal is in S_b; verification signal is in S_w
- Maximize distance between classes under constraint that intrapersonal variation is constant
- Linear feature mapping

Joint deep learning

- Learn features by joint identification-verification
- Minimize intra-personal variation under constraint that the distance between classes is constant
- Hierarchical nonlinear feature extraction
- Generalization power increases with more training identities

What has been learned by DeepID2+?

Properties owned by neurons?

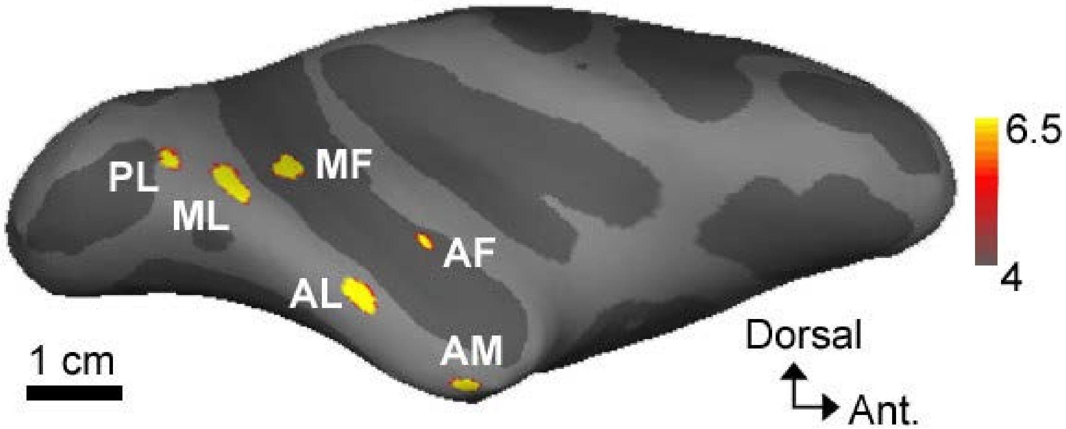
Moderate sparse

Selective to identities and attributes

Robust to data corruption

These properties are naturally owned by DeepID2+ through large-scale training, without explicitly adding regularization terms to the model

Biological Motivation

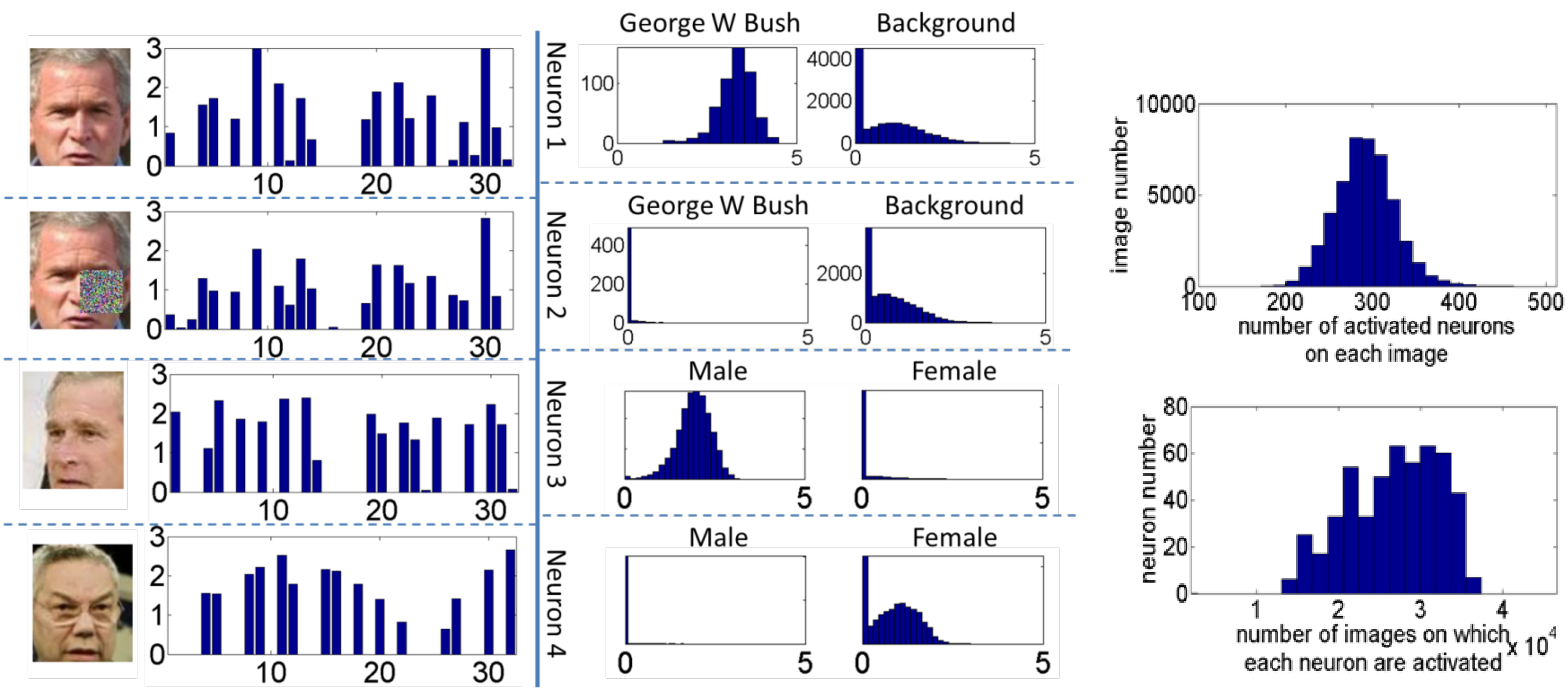


- Monkey has a face-processing network that is made of six interconnected face-selective regions
- Neurons in some of these regions were view-specific, while some others were tuned to identity across views
- View could be generalized to other factors, e.g. expressions?

Winrich A. Freiwald and Doris Y. Tsao, “Functional compartmentalization and viewpoint generalization within the macaque face-processing system,” *Science,* 330(6005):845–851, 2010.

Deeply learned features are moderately space

- For an input image, about half of the neurons are activated
- An neuron has response on about half of the images



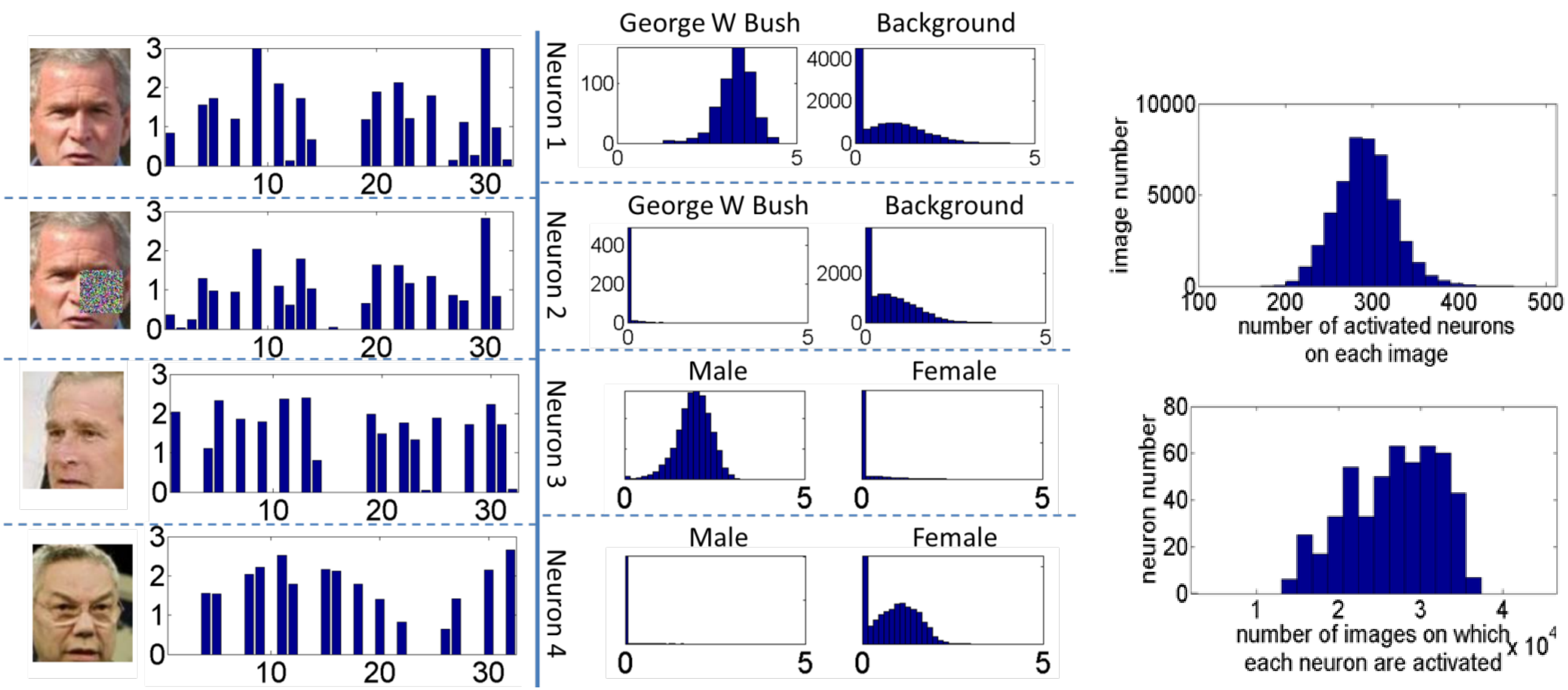
Deeply learned features are moderately space

- The binary codes on activation patterns of neurons are very effective on face recognition
- Activation patterns are more important than activation magnitudes in face recognition

	Joint Bayesian (%)	Hamming distance (%)
Single model (real values)	98.70	n/a
Single model (binary code)	97.67	96.46
Combined model (real values)	99.47	n/a
Combined model (binary code)	99.12	97.47

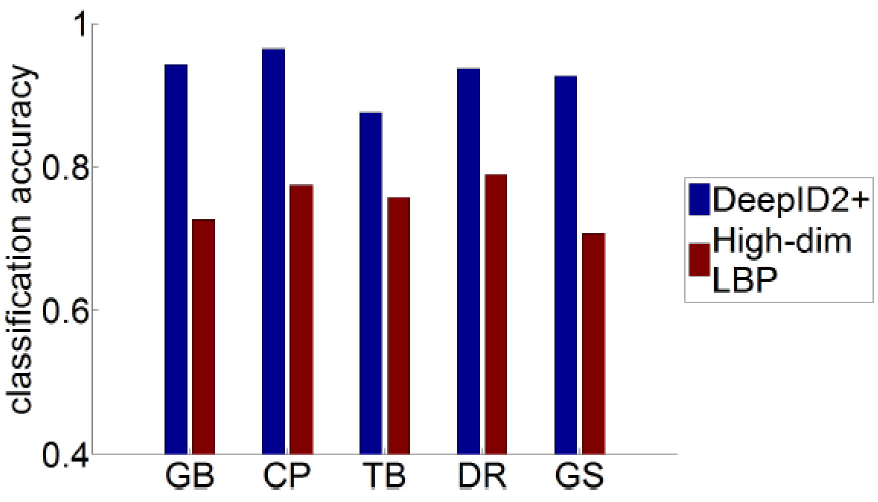
Deeply learned features are selective to identities and attributes

- With a single neuron, DeepID2 reaches 97% recognition accuracy for some identity and attribute

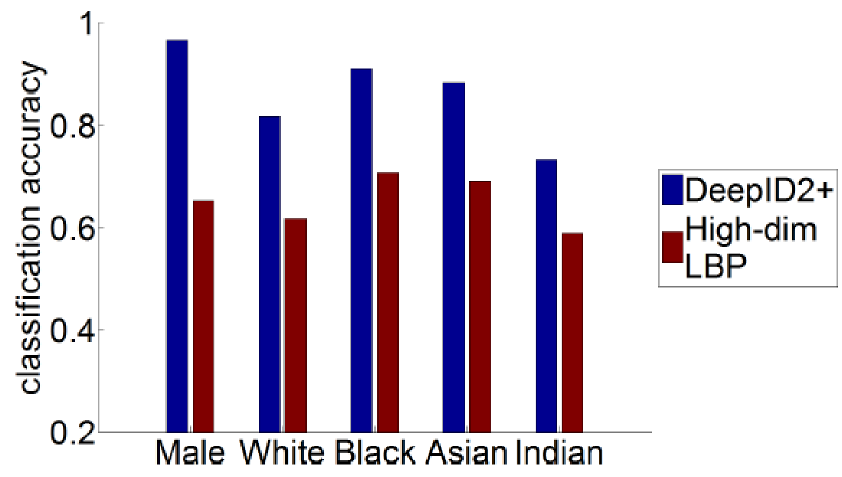


Deeply learned features are selective to identities and attributes

- With a single neuron, DeepID2 reaches 97% recognition accuracy for some identity and attribute



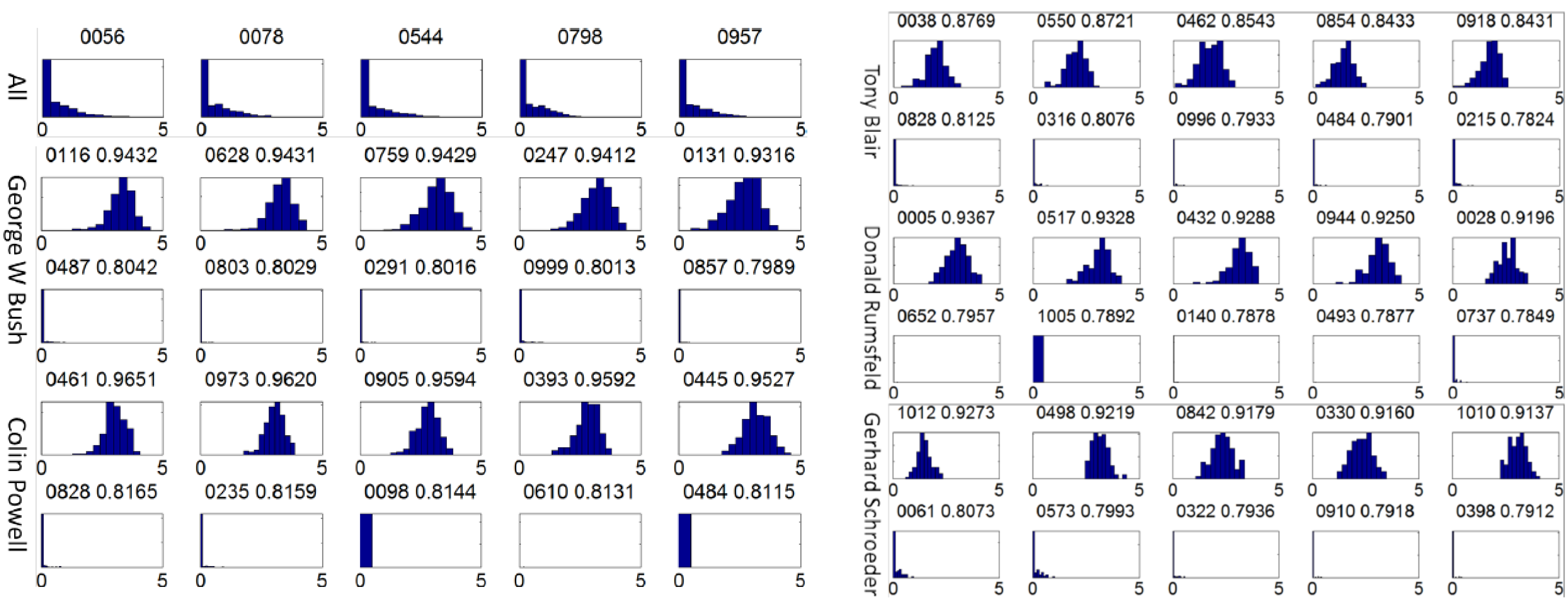
Identity classification accuracy on LFW with one single DeepID2+ or LBP feature. GB, CP, TB, DR, and GS are five celebrities with the most images in LFW.



Attribute classification accuracy on LFW with one single DeepID2+ or LBP feature.

Deeply learned features are selective to identities and attributes

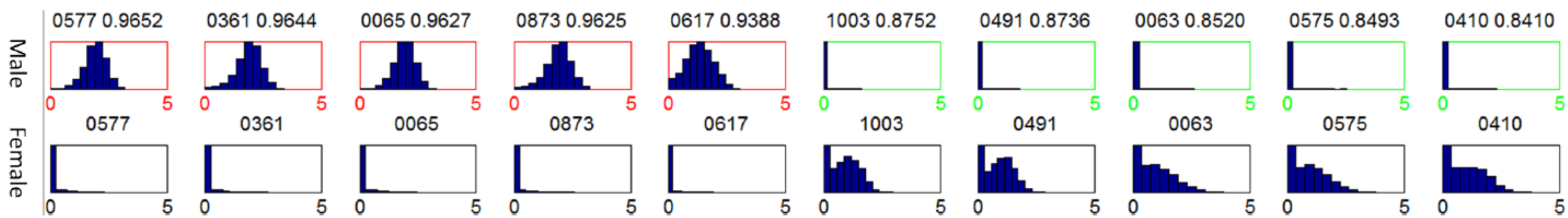
- Excitatory and inhibitory neurons



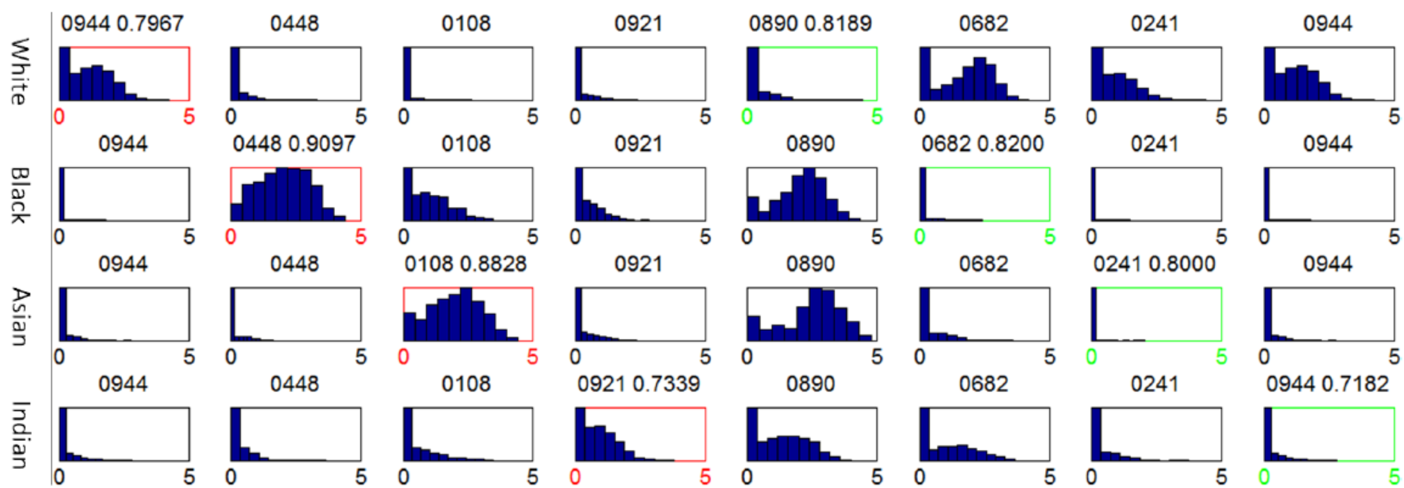
Histograms of neural activations over identities with the most images in LFW

Deeply learned features are selective to identities and attributes

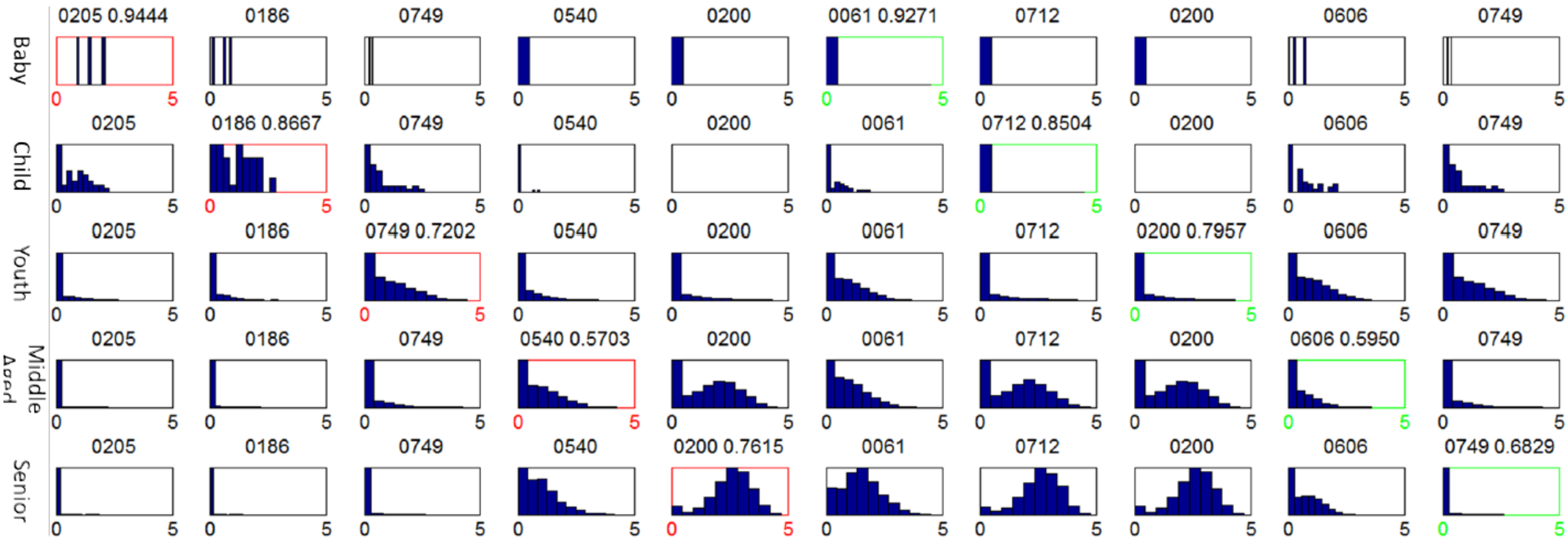
- Excitatory and inhibitory neurons



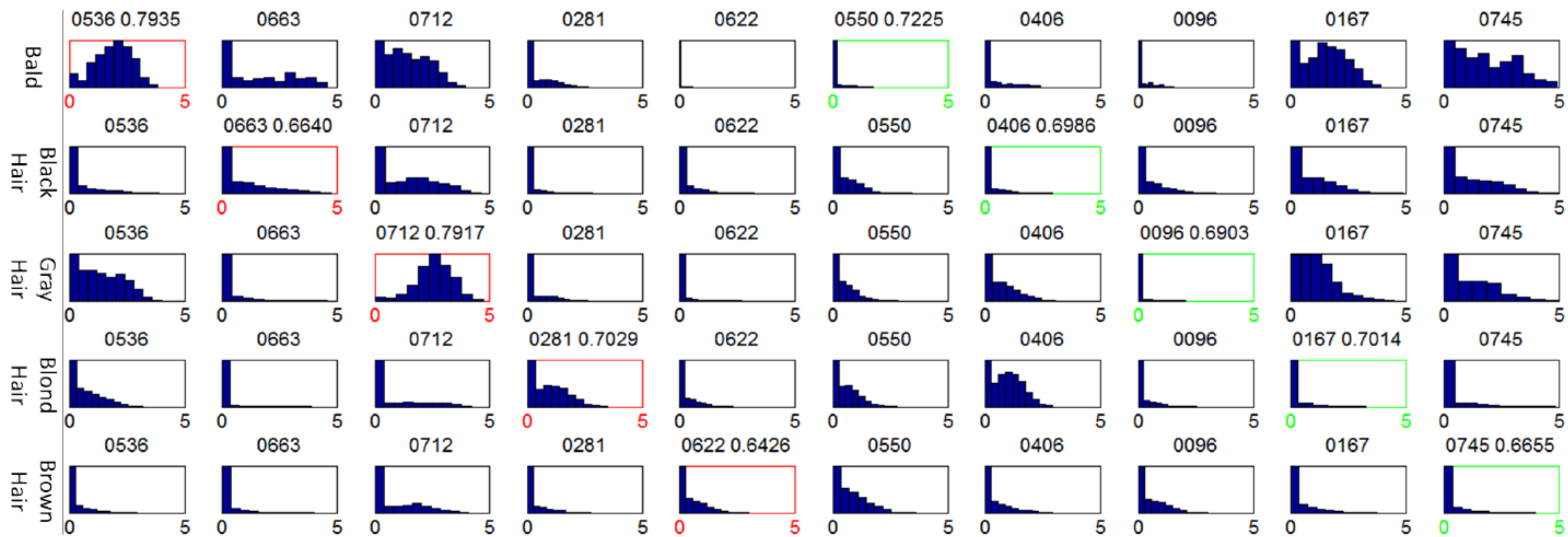
Histograms of neural activations over gender-related attributes (Male and Female)



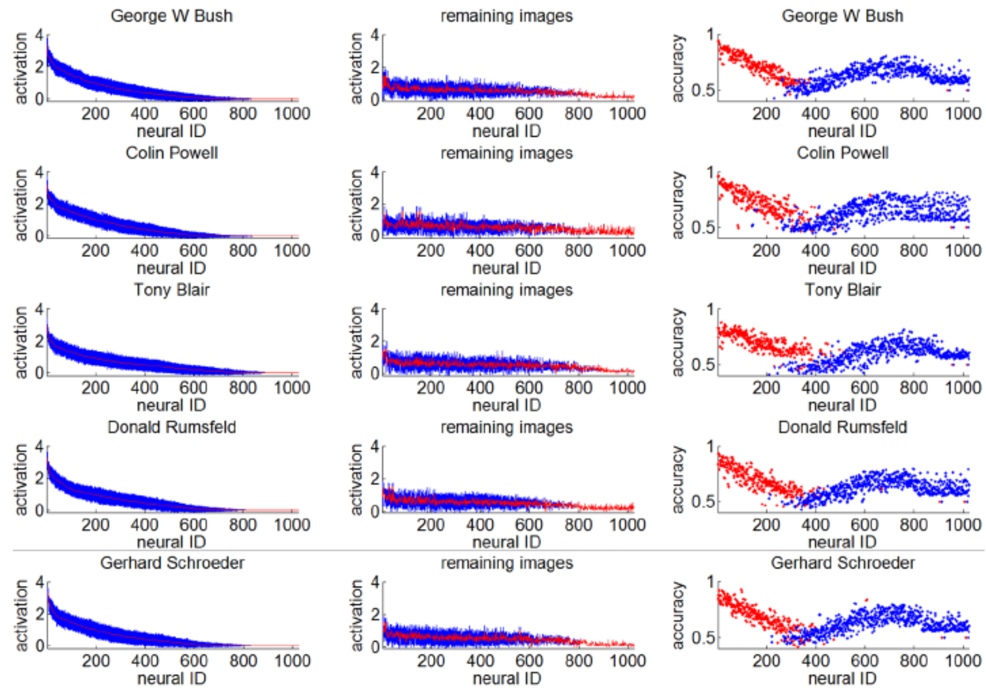
Histograms of neural activations over race-related attributes (White, Black, Asian and India)



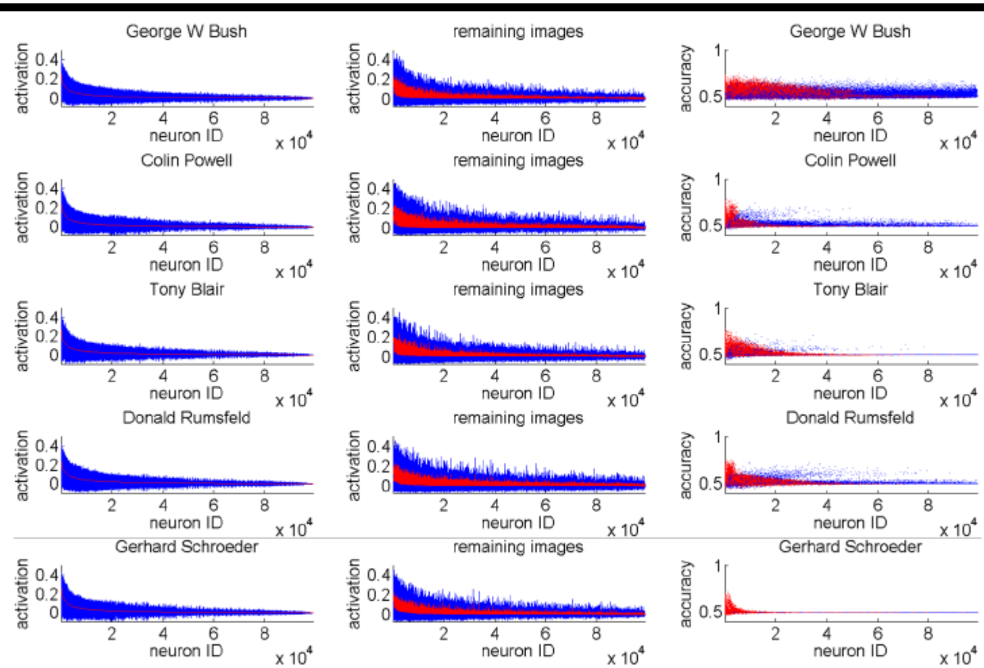
Histogram of neural activations over age-related attributes (Baby, Child, Youth, Middle Aged, and Senior)



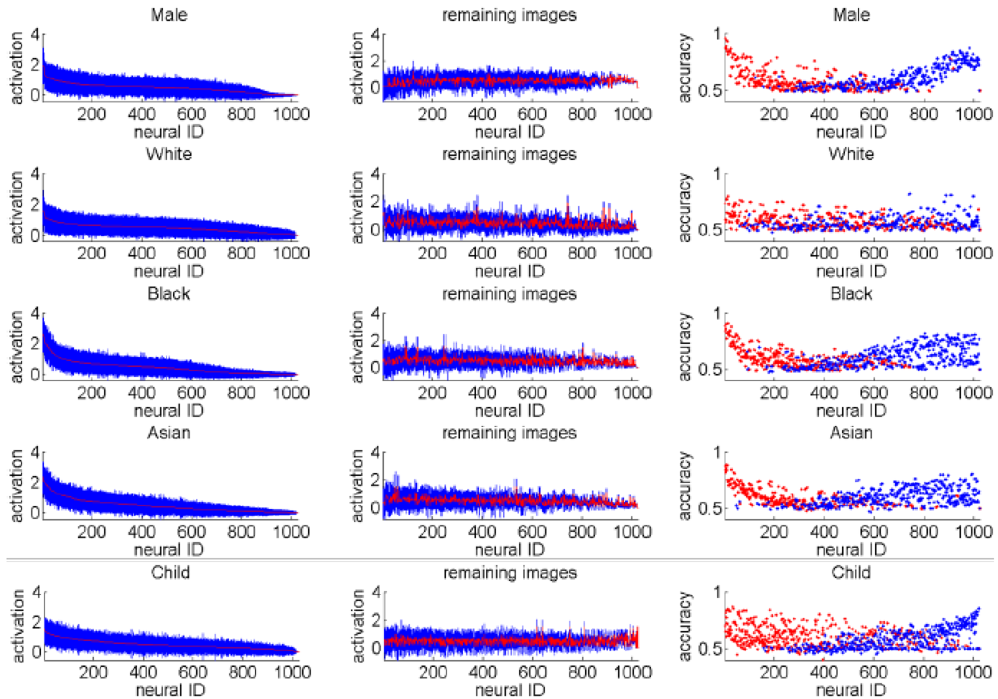
Histogram of neural activations over hair-related attributes (Bald, Black Hair, Gray Hair, Blond Hair, and Brown Hair.



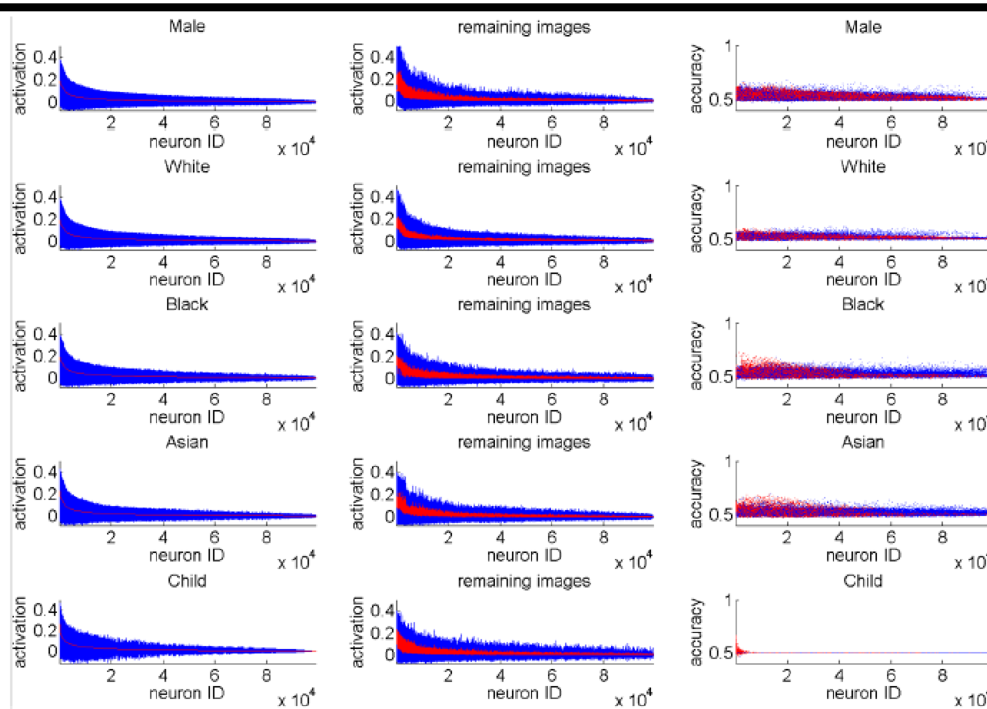
DeepID2+



High-dim LBP



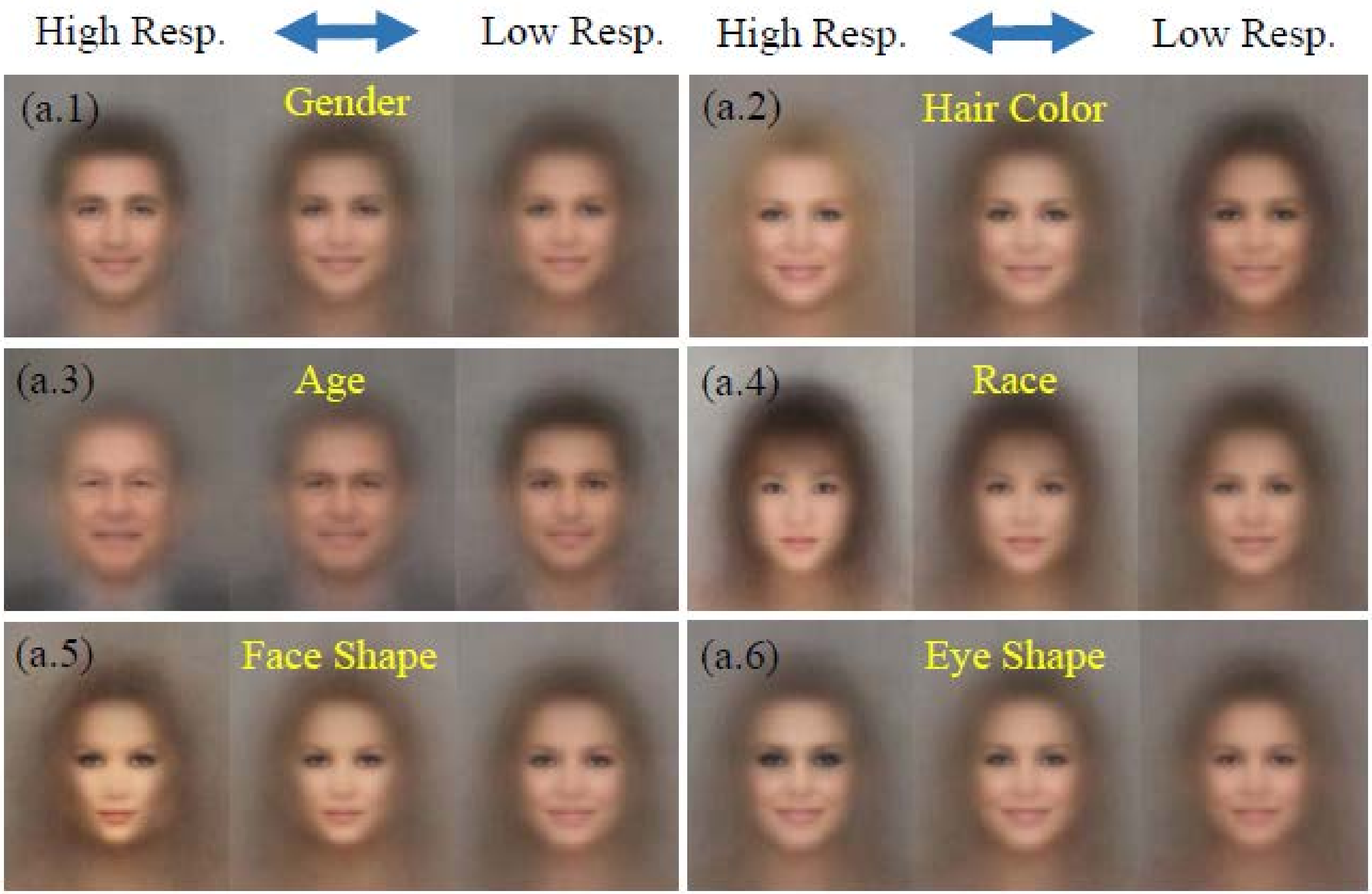
DeepID2+



High-dim LBP

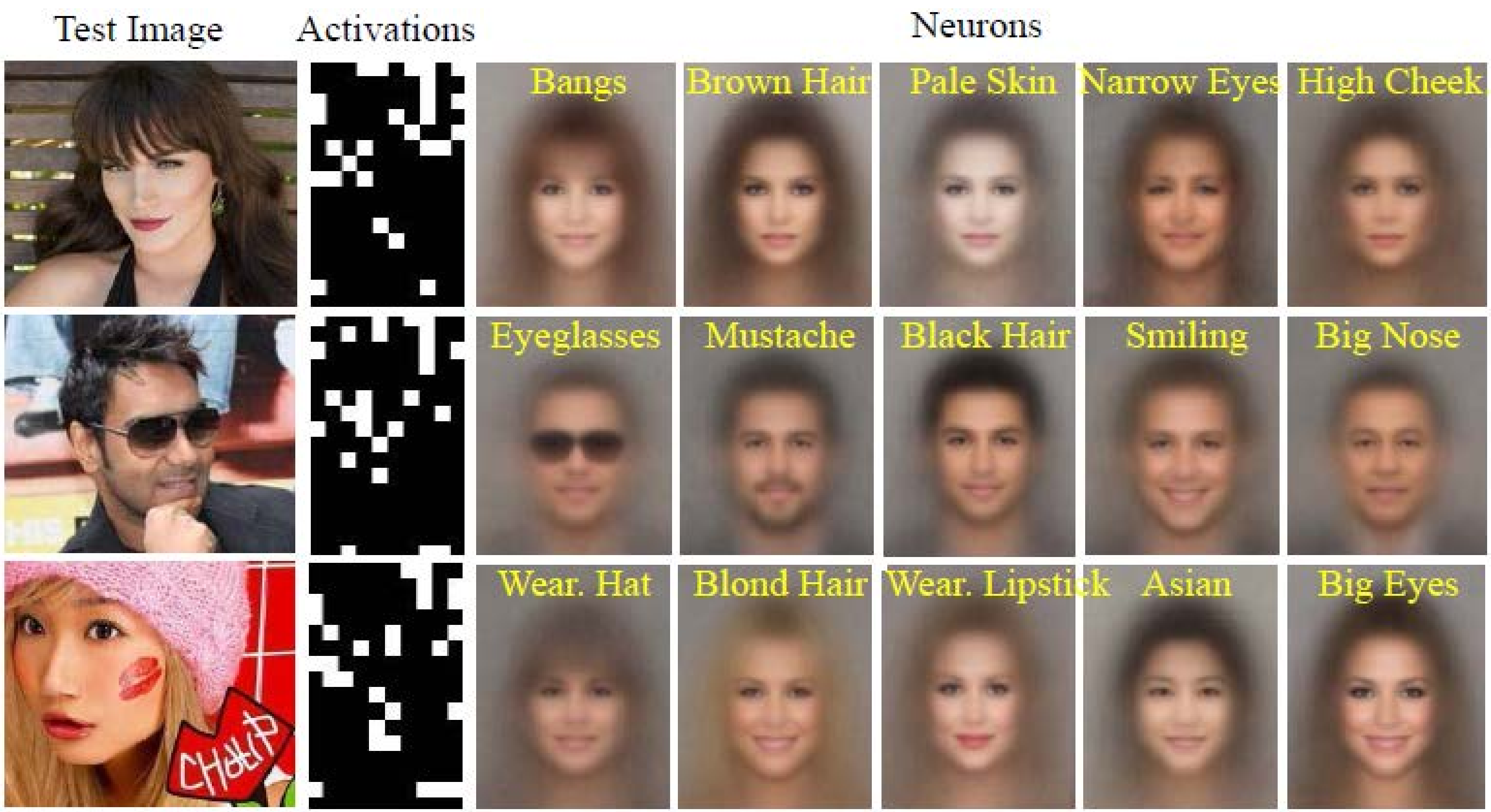
Deeply learned features are selective to identities and attributes

- Visualize the semantic meaning of each neuron



Deeply learned features are selective to identities and attributes

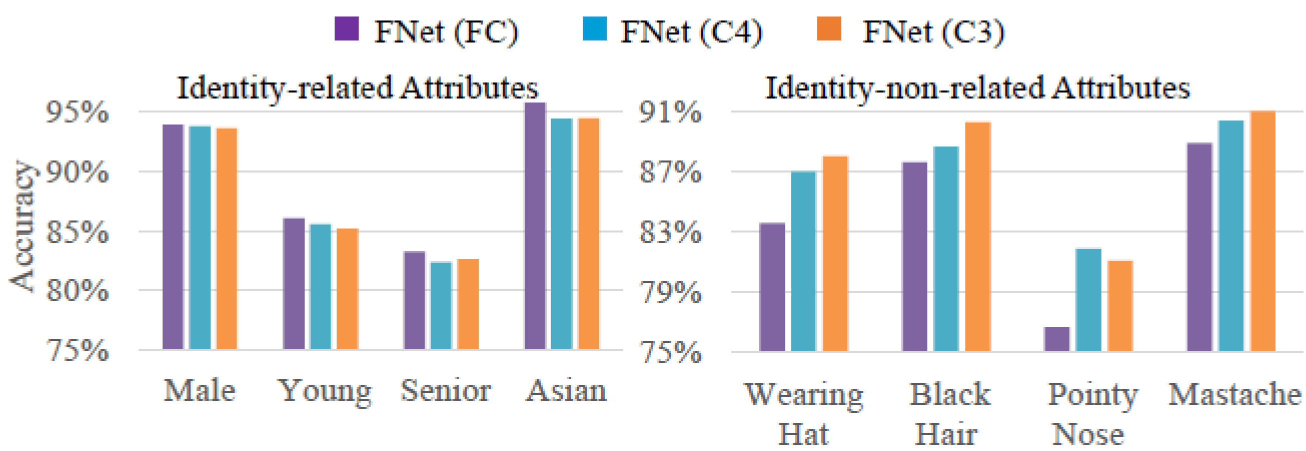
- Visualize the semantic meaning of each neuron



Neurons are ranked by their responses in descending order with respect to test images

DeepID2 features for attribute recognition

- Features at top layers are more effective on recognizing identity related attributes
- Features at lowers layers are more effective on identity-non-related attributes



DeepID2 features for attribute recognition

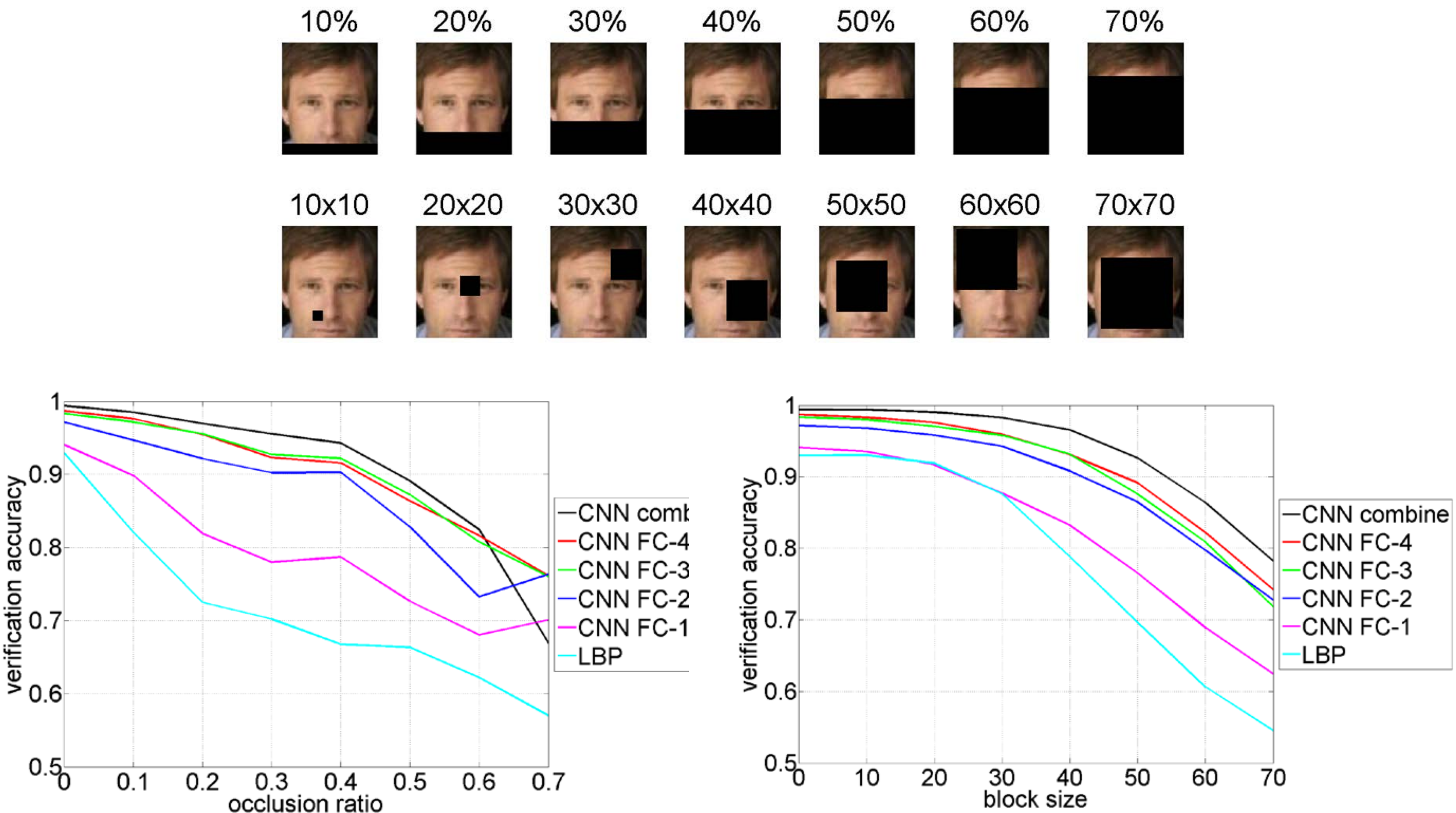
- DeepID2 features can be directly used for attribute recognition
- Use DeeID2 features as initialization (pre-trained result), and then fine tune on attribute recognition
- Average accuracy on 40 attributes on CelebA and LFWA datasets

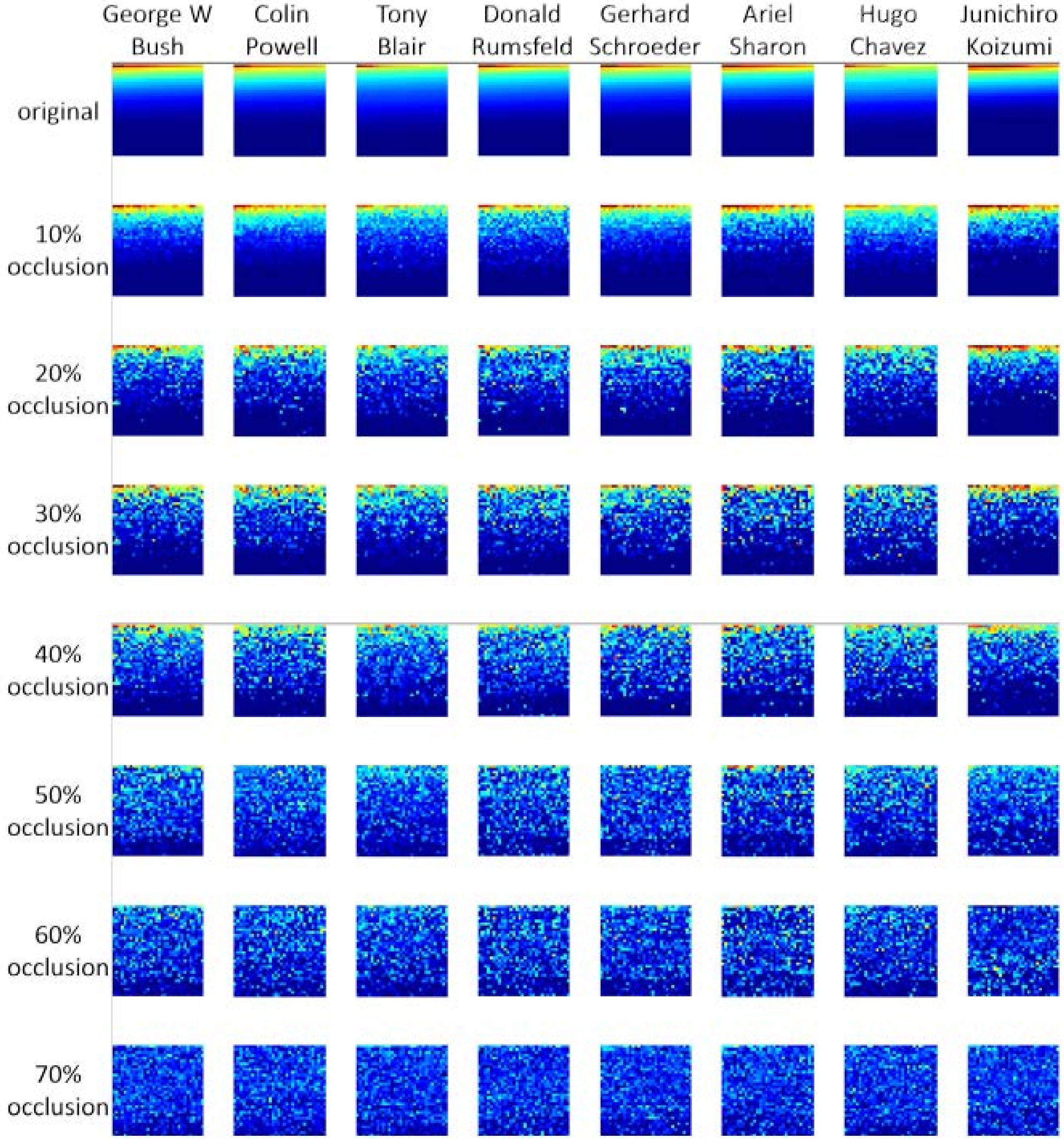
	CelebA	**LFWA**
FaceTracer [1] (HOG+SVM)	81	74
PANDA-W [2] (Parts are automatically detected)	79	71
PANDA-L [2] (Parts are given by ground truth)	85	81
DeepID2	**84**	**82**
Fine-tune (w/o DeepID2)	83	79
DeepID2 + fine-tune	**87**	**84**

Z. Liu, P. Luo, X. Wang, and X. Tang, “Deep Learning Face Attributes in the Wild,” arXiv:1411.7766, 2014.

Deeply learned features are robust to occlusions

- Global features are more robust to occlusions



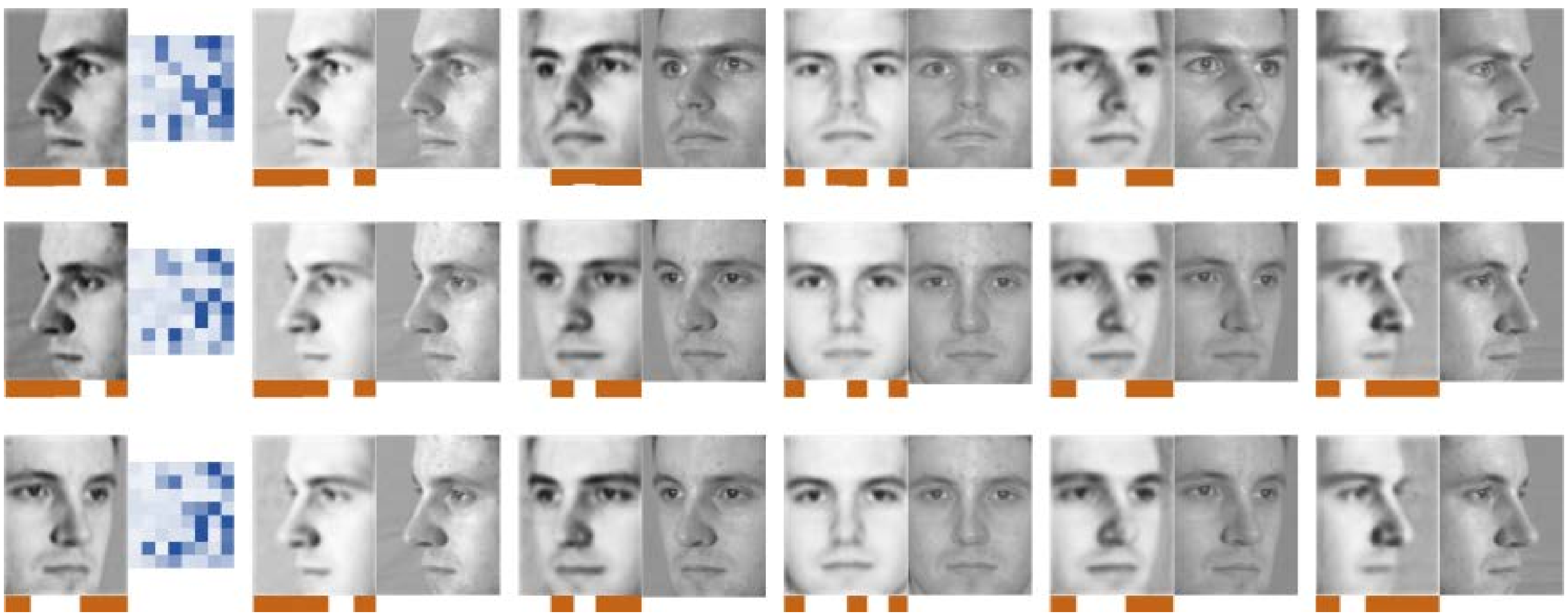

George W Bush
Colin Powell
Tony Blair
Donald Rumsfeld
Gerhard Schroeder
Ariel Sharon
Hugo Chavez
Junichiro Koizumi
original
10% occlusion
20% occlusion
30% occlusion
40% occlusion
50% occlusion
60% occlusion
70% occlusion

Outline

- Deep learning for object recognition on ImageNet
- **Deep learning for face recognition**
  - Learn identity features from joint verification-identification signals
  - **Learn 3D face models from 2D images**

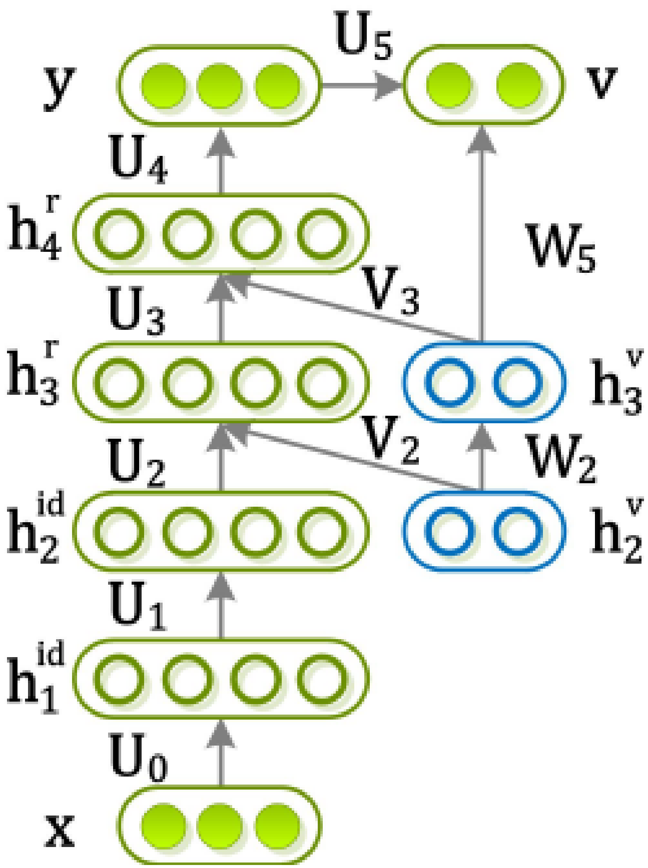
Deep Learning Multi-view Representation from 2D Images

- Inspired by brain behaviors [Winrich et al. Science 2010]
- Identity and view represented by different sets of neurons
- Given an image under arbitrary view, its viewpoint can be estimated and its full spectrum of views can be reconstructed



Z. Zhu, P. Luo, X. Wang, and X. Tang, “Deep Learning and Disentangling Face Representation by Multi-View Perception,” NIPS 2014.

Deep Learning Multi-view Representation from 2D Images



x and y are input and ouput images of the same identity but in different views;

v is the view label of the output image;

h^{id} are neurons encoding identity features

h^{v} are neurons encoding view features

h^{r} are neurons encoding features to reconstruct the output images

	Avg.	0°	-15°	$+15^\circ$	-30°	$+30^\circ$	-45°	$+45^\circ$	-60°	$+60^\circ$
Raw Pixels+LDA	36.7	81.3	59.2	58.3	35.5	37.3	21.0	19.7	12.8	7.63
LBP [1]+LDA	50.2	89.1	77.4	79.1	56.8	55.9	35.2	29.7	16.2	14.6
Landmark LBP [6]+LDA	63.2	94.9	83.9	82.9	71.4	68.2	52.8	48.3	35.5	32.1
CNN+LDA	58.1	64.6	66.2	62.8	60.7	63.6	56.4	57.9	46.4	44.2
FIP [28]+LDA	72.9	94.3	91.4	90.0	78.9	82.5	66.1	62.0	49.3	42.5
RL [28]+LDA	70.8	94.3	90.5	89.8	77.5	80.0	63.6	59.5	44.6	38.9
MTL+RL+LDA	**74.8**	**93.8**	**91.7**	**89.6**	**80.1**	**83.3**	**70.4**	**63.8**	51.5	50.2
MVP$_{\mathbf{h}_1^{id}}$+LDA	61.5	92.5	85.4	84.9	64.3	67.0	51.6	45.4	35.1	28.3
MVP$_{\mathbf{h}_2^{id}}$+LDA	**79.3**	**95.7**	**93.3**	**92.2**	**83.4**	**83.9**	**75.2**	**70.6**	**60.2**	**60.0**
MVP$_{\mathbf{h}_3^{r}}$+LDA	72.6	91.0	86.7	84.1	74.6	74.2	68.5	**63.8**	**55.7**	**56.0**
MVP$_{\mathbf{h}_4^{r}}$+LDA	62.3	83.4	77.3	73.1	62.0	63.9	57.3	53.2	44.4	46.9

Face recognition accuracies across views and illuminations on the Multi-PIE dataset. The first and the second best performances are in bold.

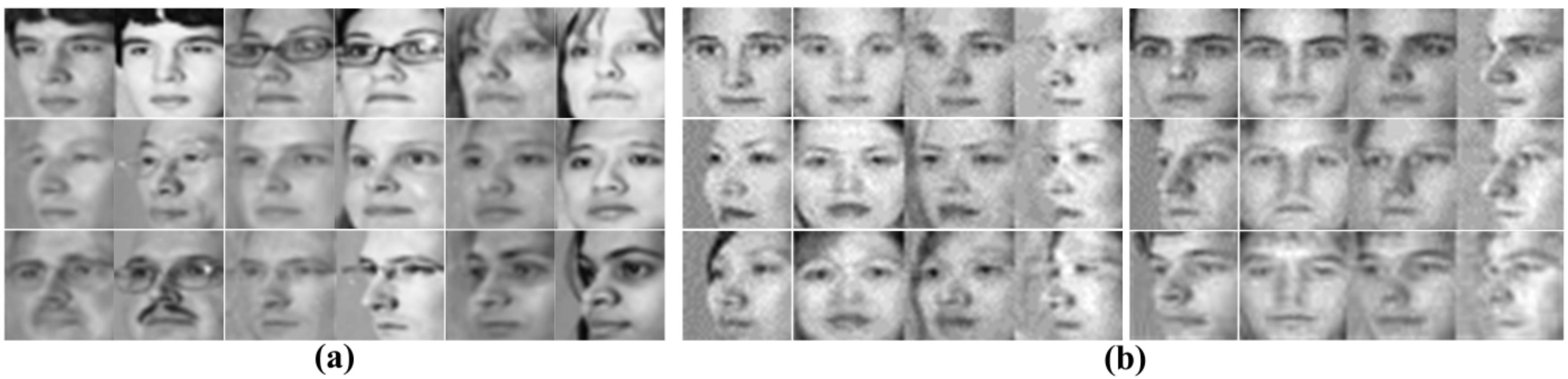
[1] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. *TPAMI*, 28:2037–2041, 2006.

[6] Dong Chen, Xudong Cao, Fang Wen, and Jian Sun. Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification. In *CVPR*, 2013.

[28] Z. Zhu, P. Luo, X. Wang, and X. Tang. Deep learning identity preserving face space. In *ICCV*, 2013.

Deep Learning Multi-view Representation from 2D Images

- Interpolate and predict images under viewpoints unobserved in the training set



The training set only has viewpoints of 0°, 30°, and 60°. (a): the reconstructed images under 15° and 45° when the input is taken under 0°. (b) The input images are under 15° and 45°.

Outline

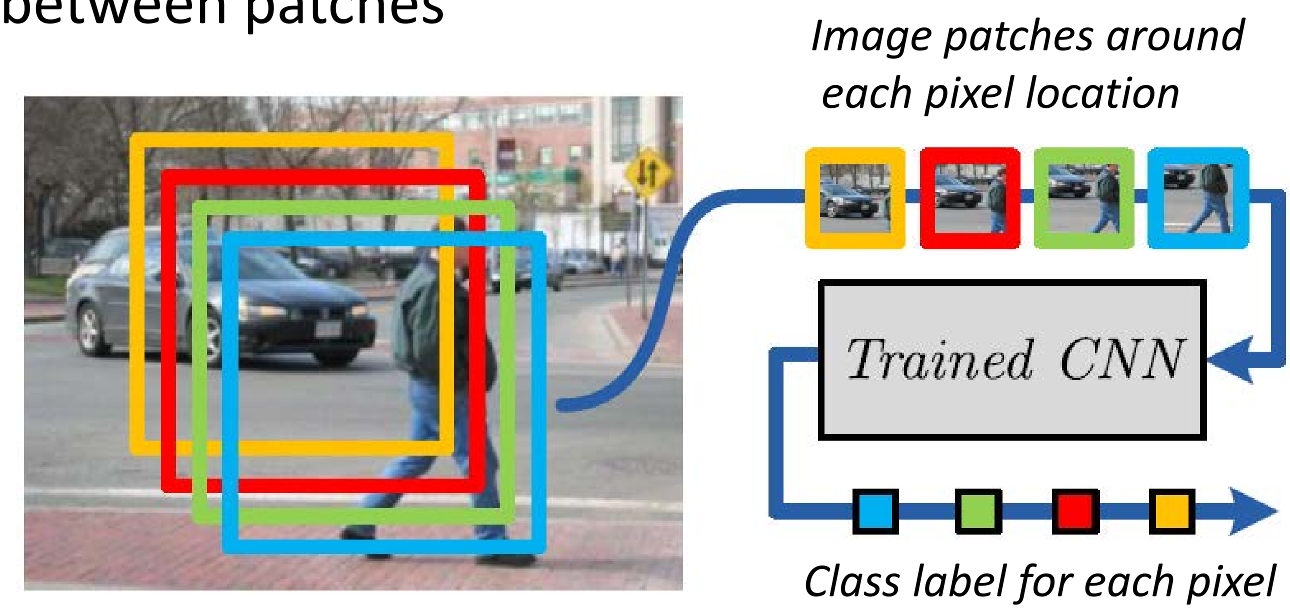
- Introduction to deep learning
- Deep learning for object recognition
- **Deep learning for object segmentation**
- Deep learning for object detection
- Open questions and future works

Whole-image classification vs pixelwise classification

- Whole-image classification: predict a single label for the whole image
- Pixelwise classification: predict a label at every pixel
  - Segmentation, detection, and tracking
- CNN, forward and backward propagation were originally proposed for whole-image classification
- Such difference was ignored when CNN was applied to pixelwise classification problems, therefore it encountered efficiency problems

Pixelwise Classification

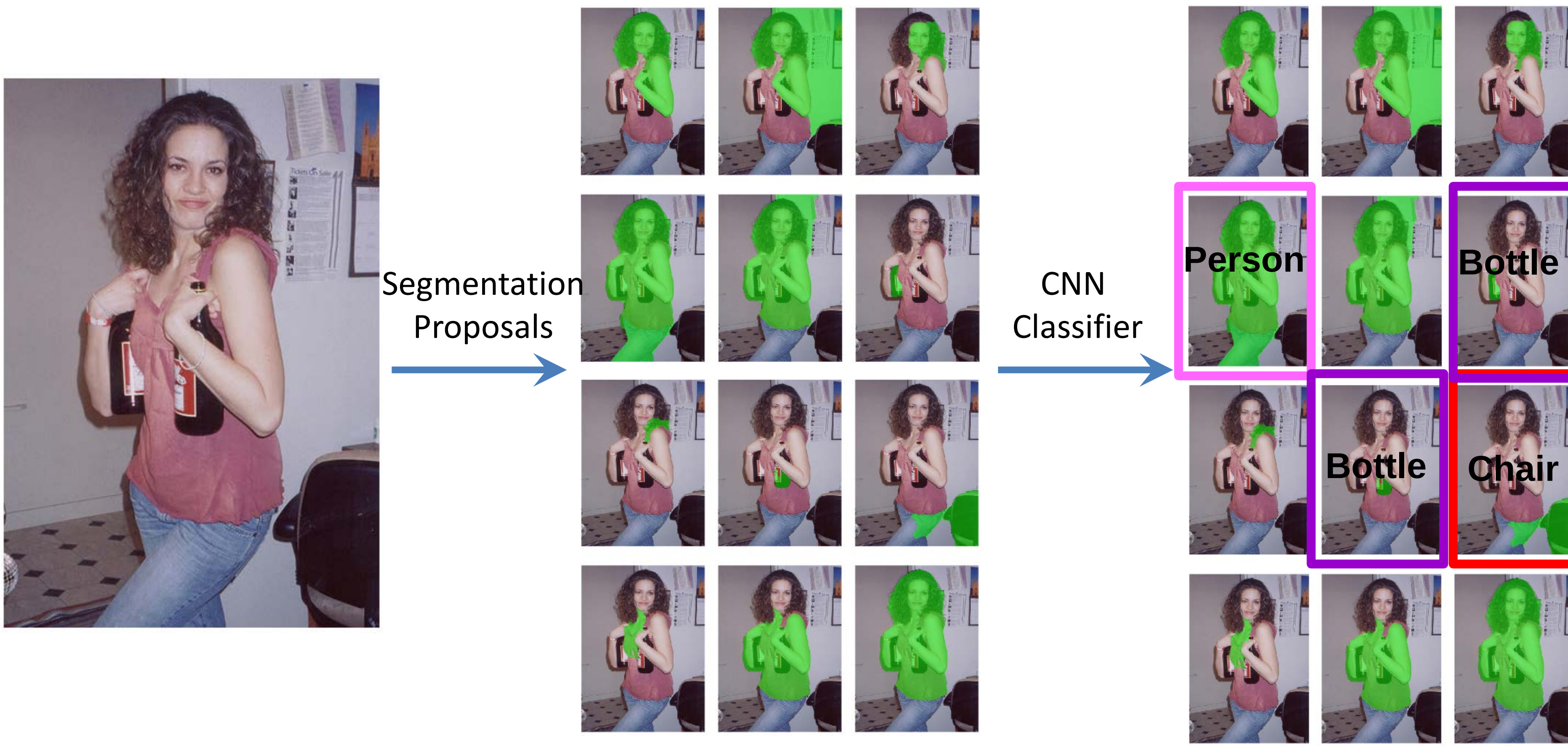
- Image patches centered at each pixel are used as the input of a CNN, and the CNN predicts a class label for each pixel
  - A lot of redundant computation because of overlap between patches



Farabet et al. TPAMI 2013   Pinheiro and Collobert ICML 2014

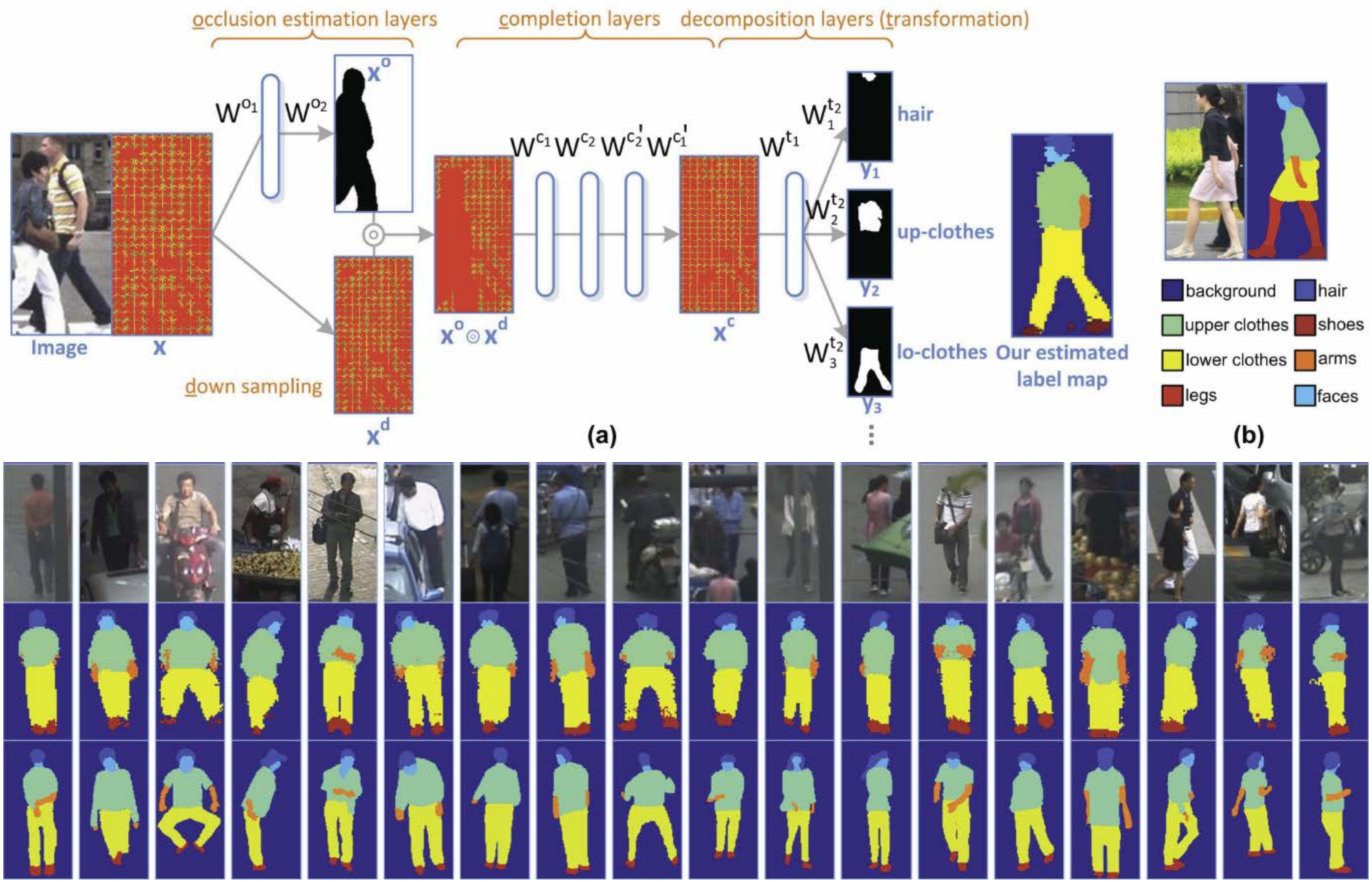
Classify Segmentation Proposal

- Determines which segmentation proposal can best represent objects on interest



R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation" CVPR 2014

Direct Predict Segmentation Maps



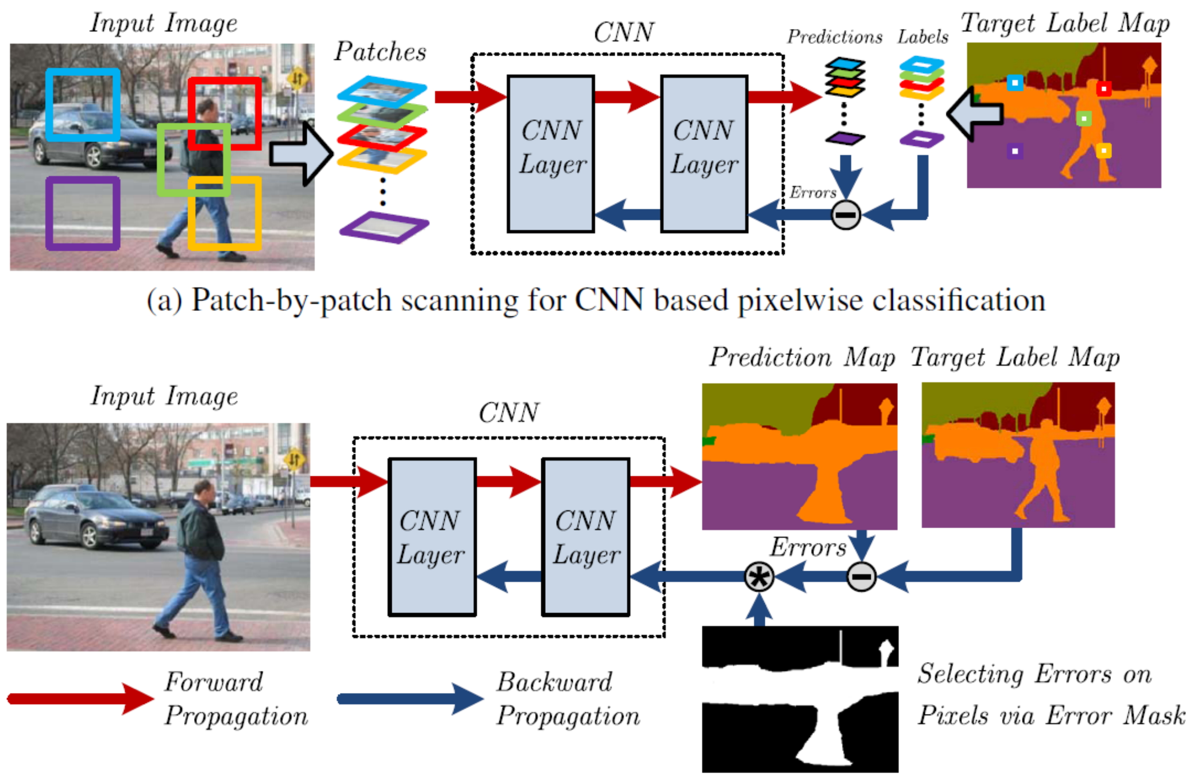
P. Luo, X. Wang, and X. Tang, "Pedestrian Parsing via Deep Decompositional Network," ICCV 2013.

Direct Predict Segmentation Maps

- Classifier is location sensitive has no translation invariance
  - Prediction not only depends on the neighborhood of the pixel, but also its location
- Only suitable for images with regular structures, such as faces and humans

Efficient Forward-Propagation of Convolutional Neural Networks

- Generate the same result as patch-by-patch scanning, with 1500 times speedup for both forward and backward propagation



(a) Patch-by-patch scanning for CNN based pixelwise classification

(b) Our approach

H. Li, R. Zhao, and X. Wang, "Highly Efficient Forward and Backward Propagation of Convolutional Neural Networks for Pixelwise Classification," arXiv:1412.4526, 2014

Speedup = $O(s^2 m^2 / (s+m)^2)$ s^2 is image size and m^2 is patch size

Layer Type	conv11	pool11	tanh11	conv12	conv13	conv21	pool21	tanh21
Kernel Size / Stride	$25 \times 8 \times 8/1$	$2 \times 2/2$	-	$50 \times 8 \times 8/1$	$32 \times 1 \times 1/1$	$25 \times 8 \times 8/1$	$2 \times 2/2$	-
Sliding Window Fwd. Prop. (ms)	39485.6	1960.2	693.0	59017.2	6473.1	63548.4	332.2	98.14
Our Method Fwd. Prop. (ms)	4.398	0.854	0.337	24.42	2.466	28.90	0.70	0.227
Speedup by Ours Fwd. Prop.	8978.1	2295.3	2056.4	2416.8	2631.3	2198.9	474.6	426.7
Sliding Window Bwd. Prop. (ms)	73961.5	10054.8	602.6	146019.3	25206.7	133706.2	1623.8	106.7
Our Method Bwd. Prop. (ms)	8.193	1.428	0.282	66.55	6.778	71.69	0.844	0.245
Speedup by Ours Bwd. Prop.	9027.4	7041.2	2136.9	2194.1	3718.9	1865.1	1923.9	6627.8

Layer Type	conv22	conv23	conv31	pool31	tanh31	conv32	conv33	Overall
Kernel Size / Stride	$50 \times 8 \times 8/1$	$32 \times 1 \times 1/1$	$25 \times 8 \times 8/1$	$2 \times 2/2$	-	$50 \times 8 \times 8/1$	$32 \times 1 \times 1/1$	
Sliding Window Fwd. Prop. (ms)	14765.3	2433.4	17059.8	32.15	13.81	17015.4	2069.7	224997.4
Our Method Fwd. Prop. (ms)	18.98	1.920	20.55	0.488	0.164	10.76	1.080	116.2
Speedup by Ours Bwd. Prop.	777.9	1267.4	830.2	65.9	84.2	1581.4	1916.4	1935.6
Sliding Window Bwd. Prop. (ms)	28744.1	8522.3	16727.5	128.358	15.91	8657.7	2793.6	456871.1
Our Method Fwd. Prop. (ms)	52.35	5.368	50.89	0.630	0.180	29.47	3.117	298.0
Speedup by Ours Bwd. Prop.	549.1	1587.6	328.7	203.7	88.4	293.8	896.2	1533.1

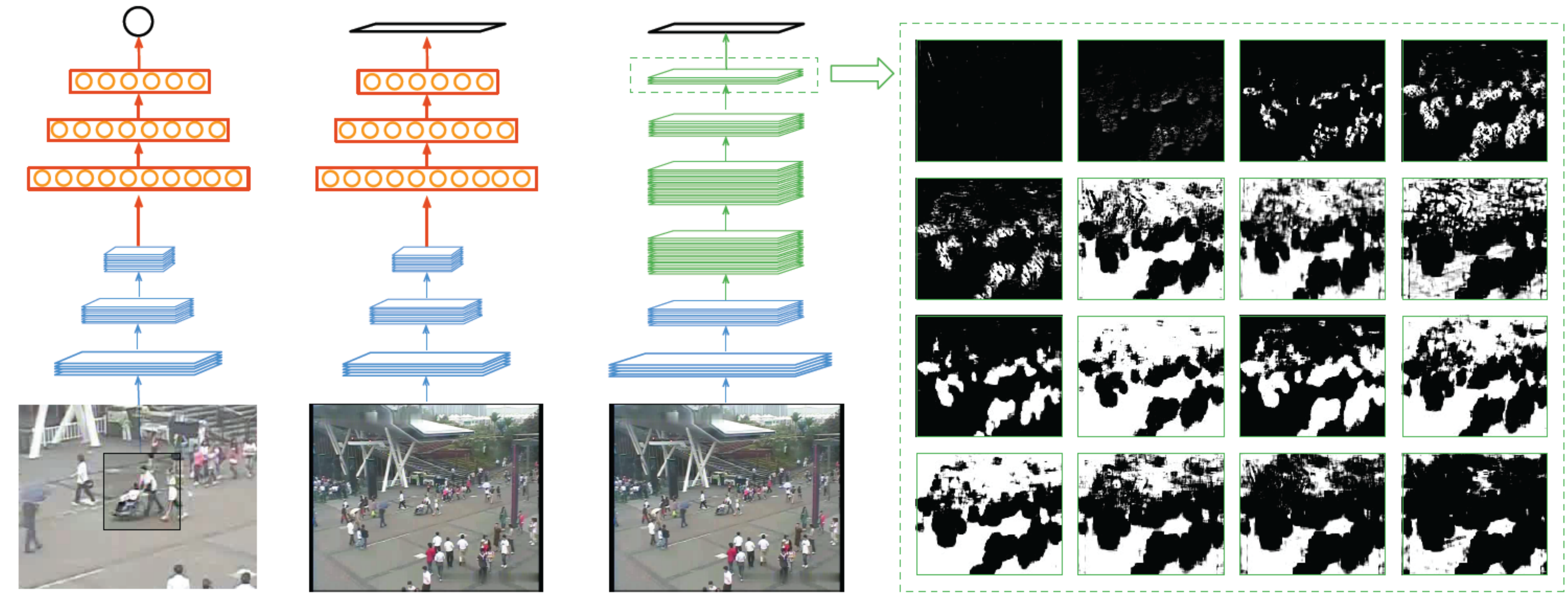
The layewise timing and speedup results of the forward and backward propagation by our proposed algorithm on the RCNN model with 3X410X410 images as inputs.

Fully convolutional neural network

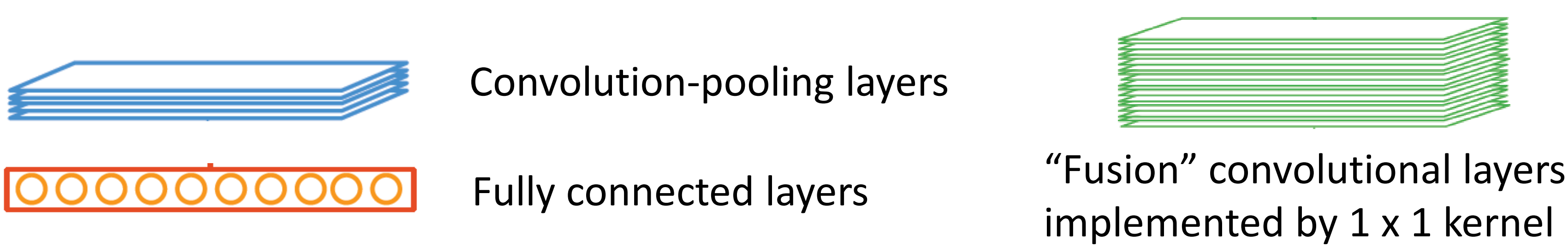
- Replace fully connected layers in CNN with 1 x 1 convolution kernel just like "network in network" (Lin, Chen and Yan, arXiv 2013)
- Take the whole images as inputs and directly output segmentation map
- Has translation invariance like patch-by-patch scanning, but with much lower computational cost
- Once FCNN is learned, it can process input images of any sizes without warping them to a standard size

K. Kang and X. Wang, "Fully Convolutional Neural Networks for Crowd Segmentation," arXiv:1411.4464, 2014

Fully convolutional neural network



(a) CNN Patch-scanning (b) CNN Regression (c) FCNN Segmentation (d) FCNN Feature Maps



Summary

- Deep learning significantly outperforms conventional vision systems on large scale image classification
- Feature representation learned from ImageNet can be well generalized to other tasks and datasets
- In face recognition, identity preserving features can be effectively learned by joint identification-verification signals
- 3D face models can be learned from 2D images; identity and pose information is encoded by different sets of neurons
- In segmentation, larger patches lead to better performance because of the large learning capacity of deep models. It is also possible to directly predict the segmentation map.
- The efficiency of CNN based segmentation can be significantly improved by considering the differences between whole-image classification and pixelwise classification

References

- A. Krizhevsky, L. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," Proc. NIPS, 2012.
- G. B. Huang, H. Lee, and E. Learned-Miller, "Learning Hierarchical Representation for Face Verification with Convolutional Deep Belief Networks," Proc. CVPR, 2012.
- Y. Sun, X. Wang, and X. Tang, "Hybrid Deep Learning for Computing Face Similarities," Proc. ICCV, 2013.
- Y. Sun, X. Wang, and X. Tang, "Deep Learning Face Representation from Predicting 10,000 classes," Proc. CVPR, 2014.
- Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," Proc. CVPR, 2014.
- Y. Sun, X. Wang, and X. Tang, "Deep Learning Face Representation by Joint Identification-Verification," NIPS, 2014.
- A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN Features off-the-shelf: an Astounding Baseline for Recognition," arXiv preprint arXiv:1403.6382, 2014.
- Y. Gong, L. Wang, R. Guo, and S. Lazebnik, "Multi-Scale Orderless Pooling of Deep Convolutional Activation Features," arXiv preprint arXiv:1403.1840, 2014.

- M. Turk and A. Pentland, “Eigenfaces for Recognition,” Journal of Cognitive Neuroscience, Vol. 3, pp. 71-86, 1991.
- P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, “Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection,” TPAMI, Vol. 19, pp. 711-720, 1997.
- B. Moghaddam, T. Jebara, and A. Pentland, “Bayesian Face Recognition,” Pattern Recognition, Vol. 33, pp. 1771-1782, 2000.
- X. Wang and X. Tang, “A Unified Framework for Subspace Face Recognition,” TPAMI, Vol. 26, pp. 1222-1228, 2004.
- Z. Zhu, P. Luo, X. Wang, and X. Tang, “Deep Learning and Disentangling Face Representation by Multi-View Perception,” NIPS 2014.
- C. Farabet, C. Couprie, L. Najman, and Y. LeCun, “Learning Hierarchical Features for Scene Labeling”, TPAMI, Vol. 35, pp. 1915-1929, 2013.
- P. O. Pinheiro and R. Collobert, “Recurrent Convolutional Neural Networks for Scene Labeling”, Proc. ICML 2014.
- R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation” CVPR 2014
- P. Luo, X. Wang, and X. Tang, “Pedestrian Parsing via Deep Decompositional Network,” ICCV 2013.
- Winrich A. Freiwald and Doris Y. Tsao, “Functional compartmentalization and viewpoint generalization within the macaque face-processing system,” *Science,* 330(6005):845–851, 2010.
- Shay Ohayon, Winrich A. Freiwald, and Doris Y. Tsao. What makes a cell face selective? the importance of contrast. *Neuron,* 74:567–581, 2013.

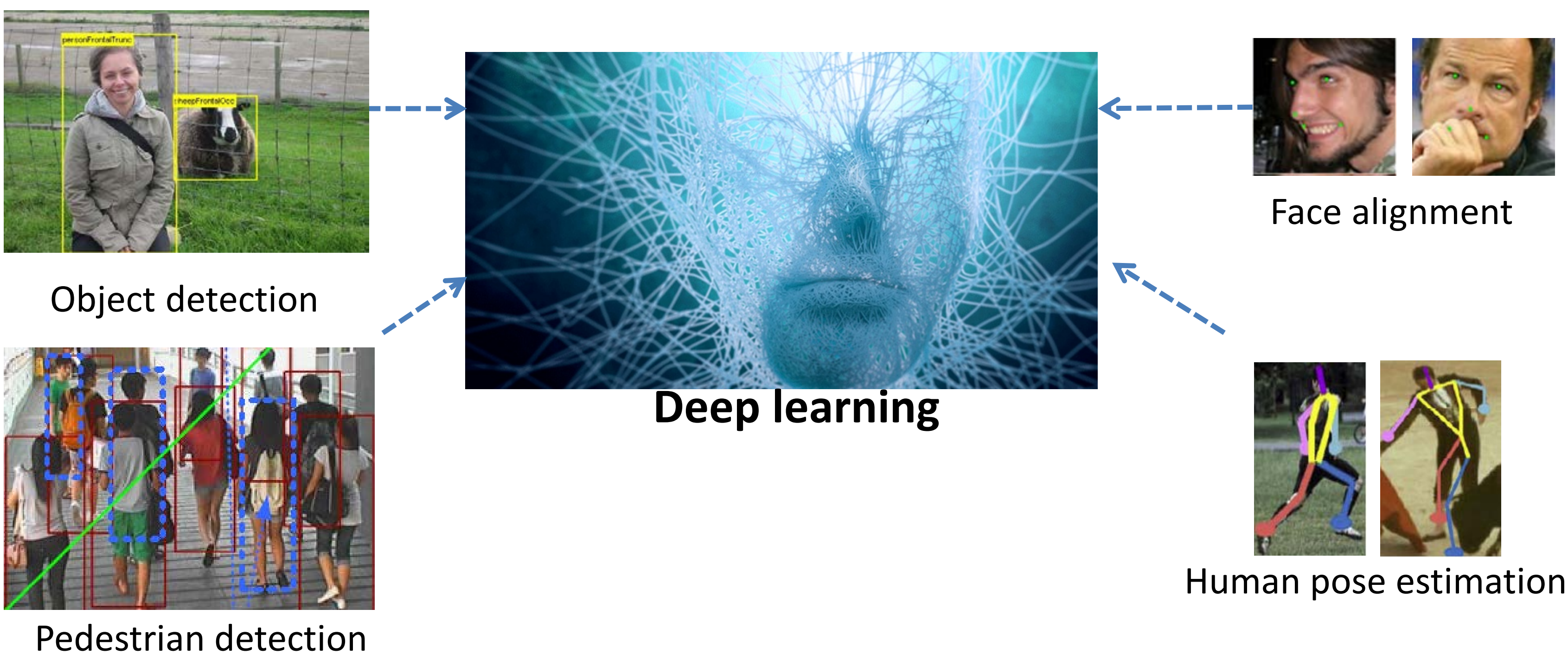
- Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. CVPR, 2015.
- Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep Learning Face Attributes in the Wild," arXiv:1411.7766, 2014.
- H. Li, R. Zhao, and X. Wang, "Highly Efficient Forward and Backward Propagation of Convolutional Neural Networks for Pixelwise Classification," arXiv:1412.4526, 2014.
- K. Kang and X. Wang, "Fully Convolutional Neural Networks for Crowd Segmentation," arXiv:1411.4464, 2014

Outline

- Introduction to deep learning
- Deep learning for object recognition
- Deep learning for object segmentation
- **Deep learning for object detection**
- Open questions and future works

Part IV: Deep Learning for Object Detection

- Pedestrian Detection
- Human part localization
- General object detection



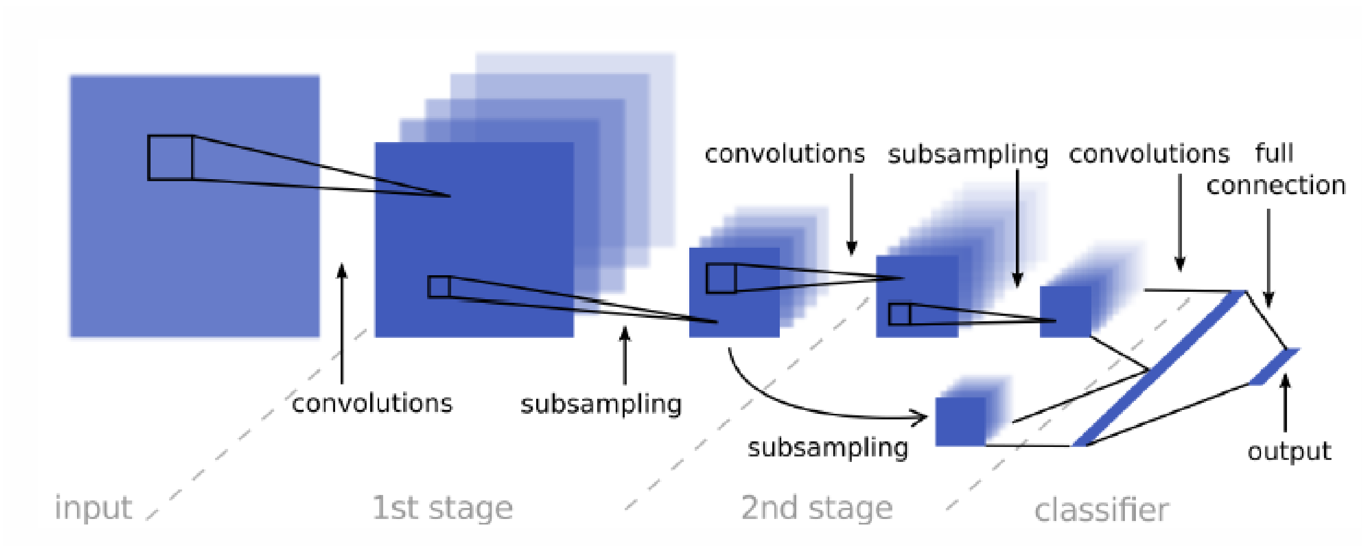
Part IV: Deep Learning for Object Detection

- Jointly optimize the detection pipeline
- Multi-stage deep learning (cascaded detectors)
- Mixture components
- Integrate segmentation and detection to depress background clutters
- Contextual modeling
- Pre-training
- Model deformation of object parts, which are shared across classes

Joint Deep Learning:

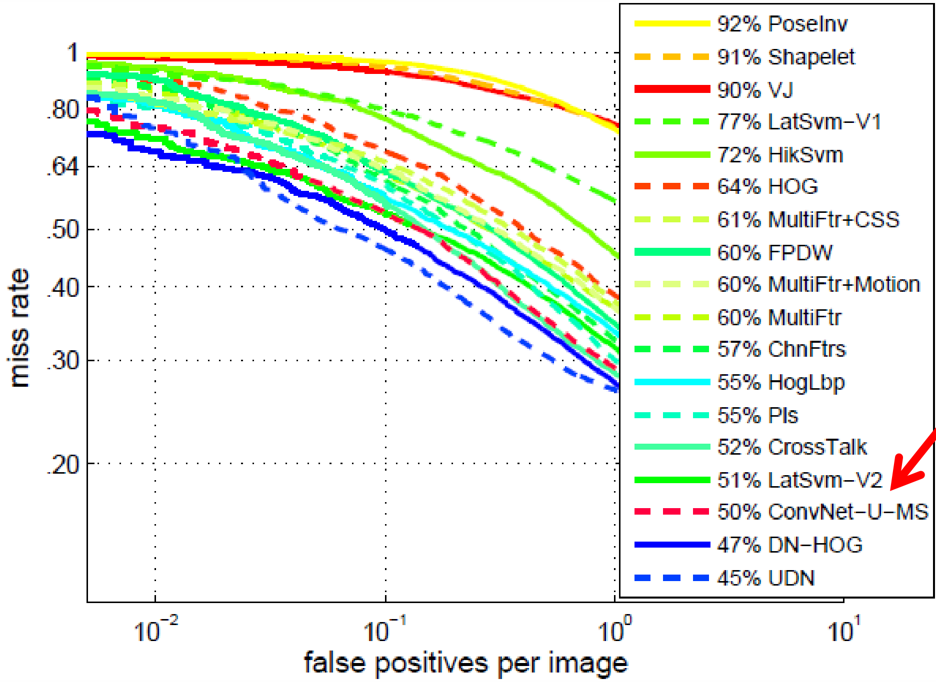
✧ **Jointly optimize the detection pipeline**

What if we treat an existing deep model as a black box in pedestrian detection?

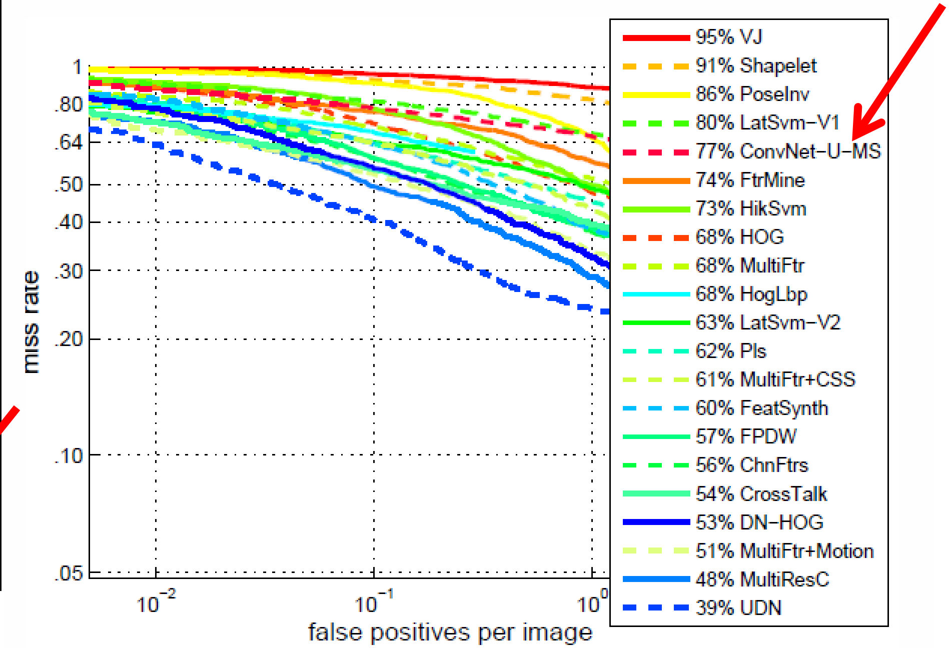


ConvNet−U−MS

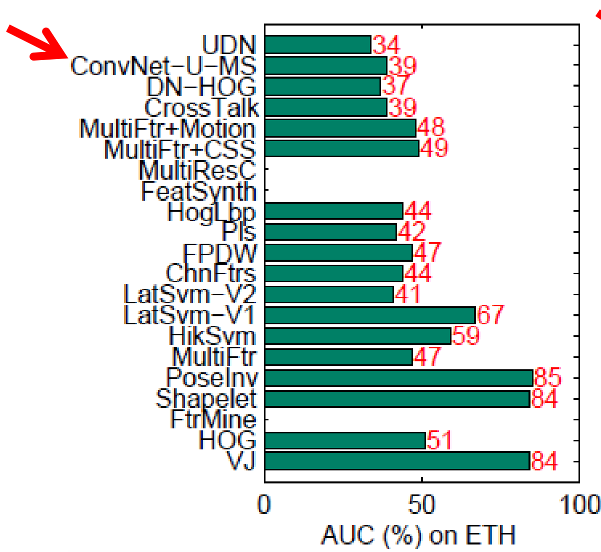
– Sermnet, K. Kavukcuoglu, S. Chintala, and LeCun, “Pedestrian Detection with Unsupervised Multi-Stage Feature Learning,” CVPR 2013.

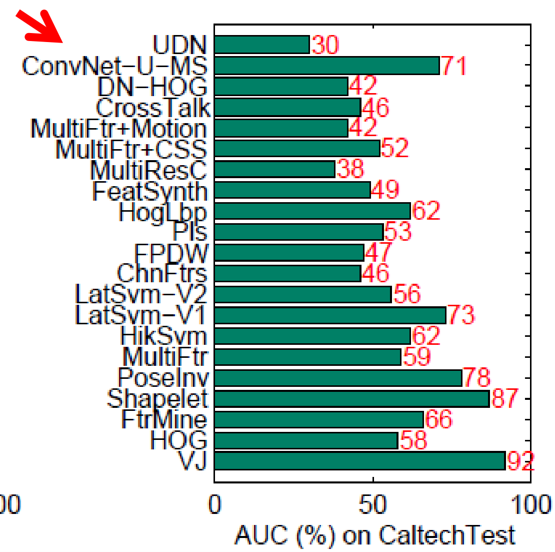


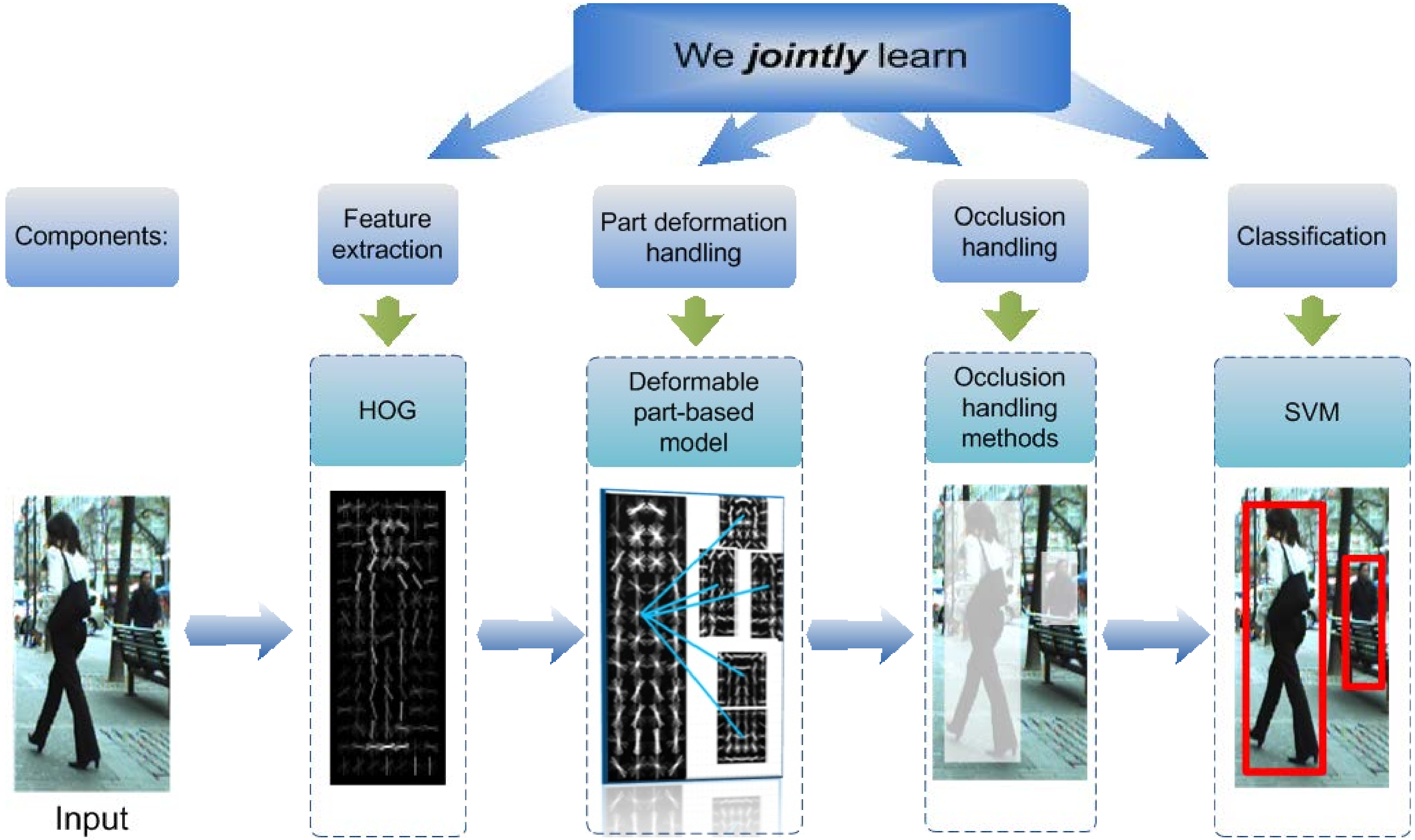
Results on ETHZ



Results on Caltech Test

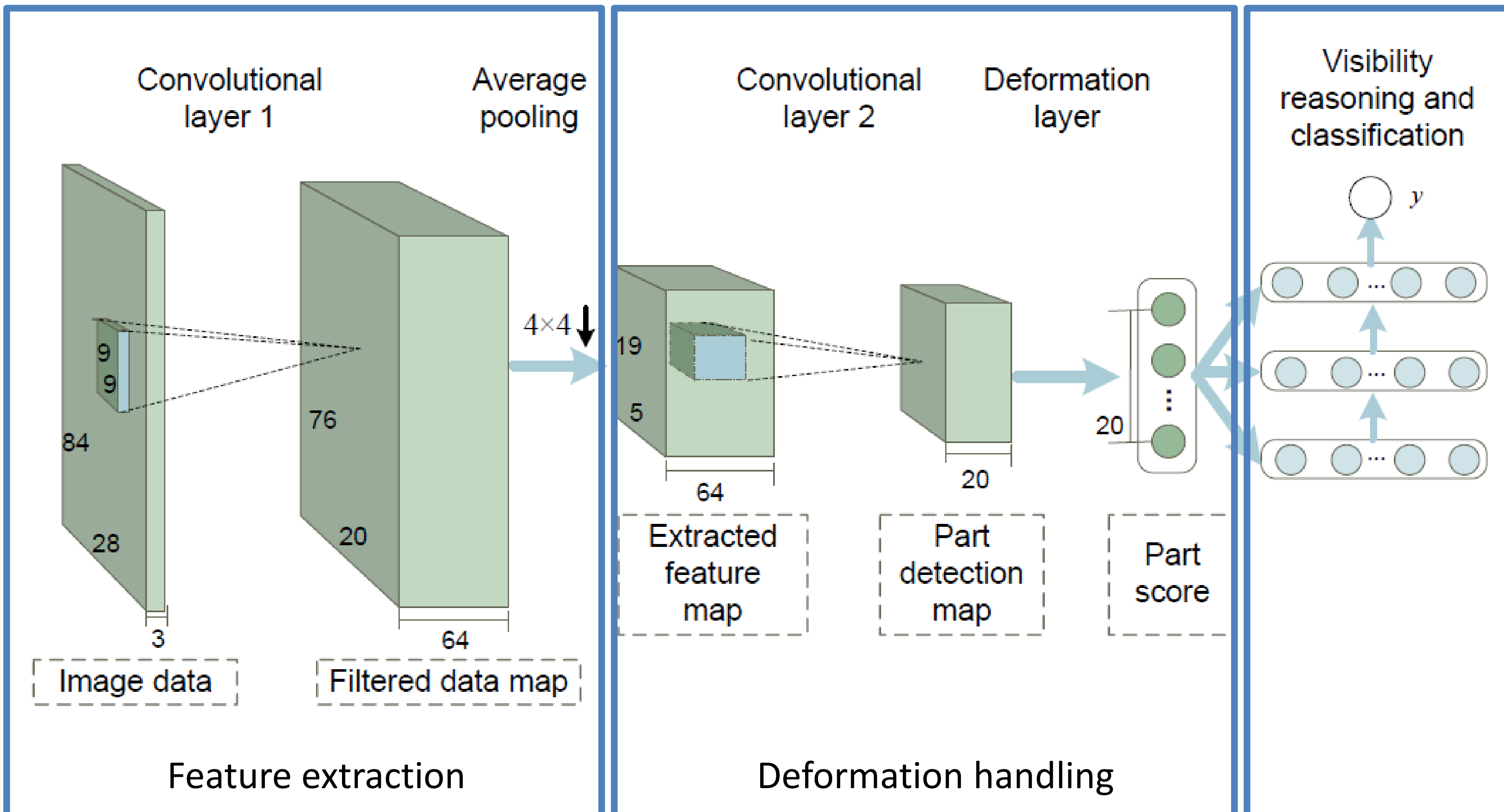






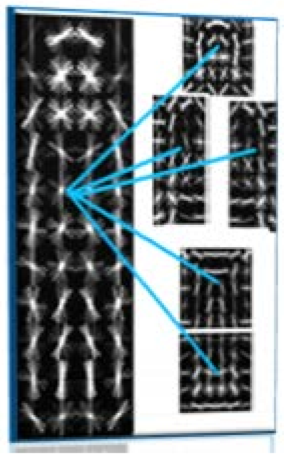
- N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. CVPR, 2005. (6000 citations)
- P. Felzenszwalb, D. McAlester, and D. Ramanan. A Discriminatively Trained, Multiscale, Deformable Part Model. CVPR, 2008. (2000 citations)
- W. Ouyang and X. Wang. A Discriminative Deep Model for Pedestrian Detection with Occlusion Handling. CVPR, 2012.

Our Joint Deep Learning Model



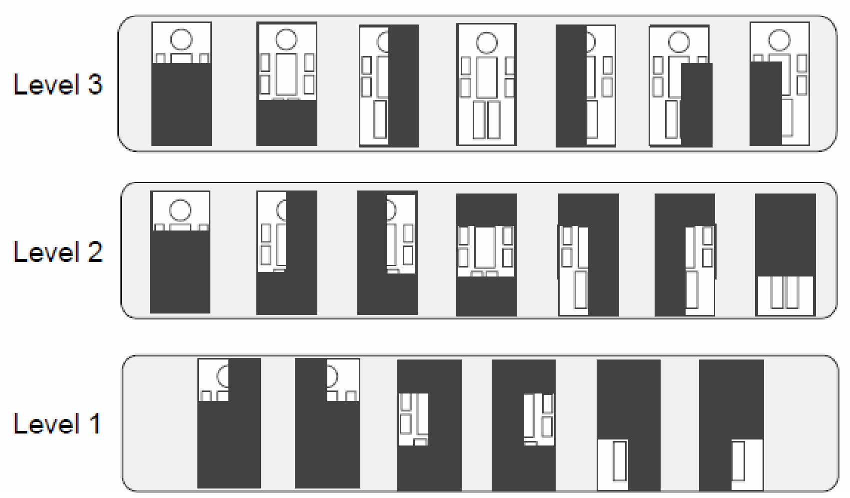
W. Ouyang and X. Wang, "Joint Deep Learning for Pedestrian Detection," Proc. ICCV, 2013.

Modeling Part Detectors

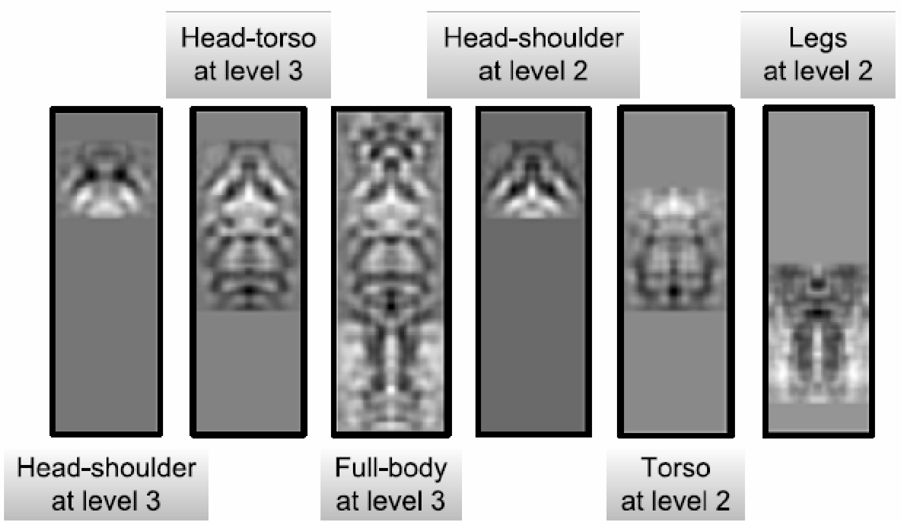


- Design the filters in the second convolutional layer with variable sizes

Part models learned from HOG

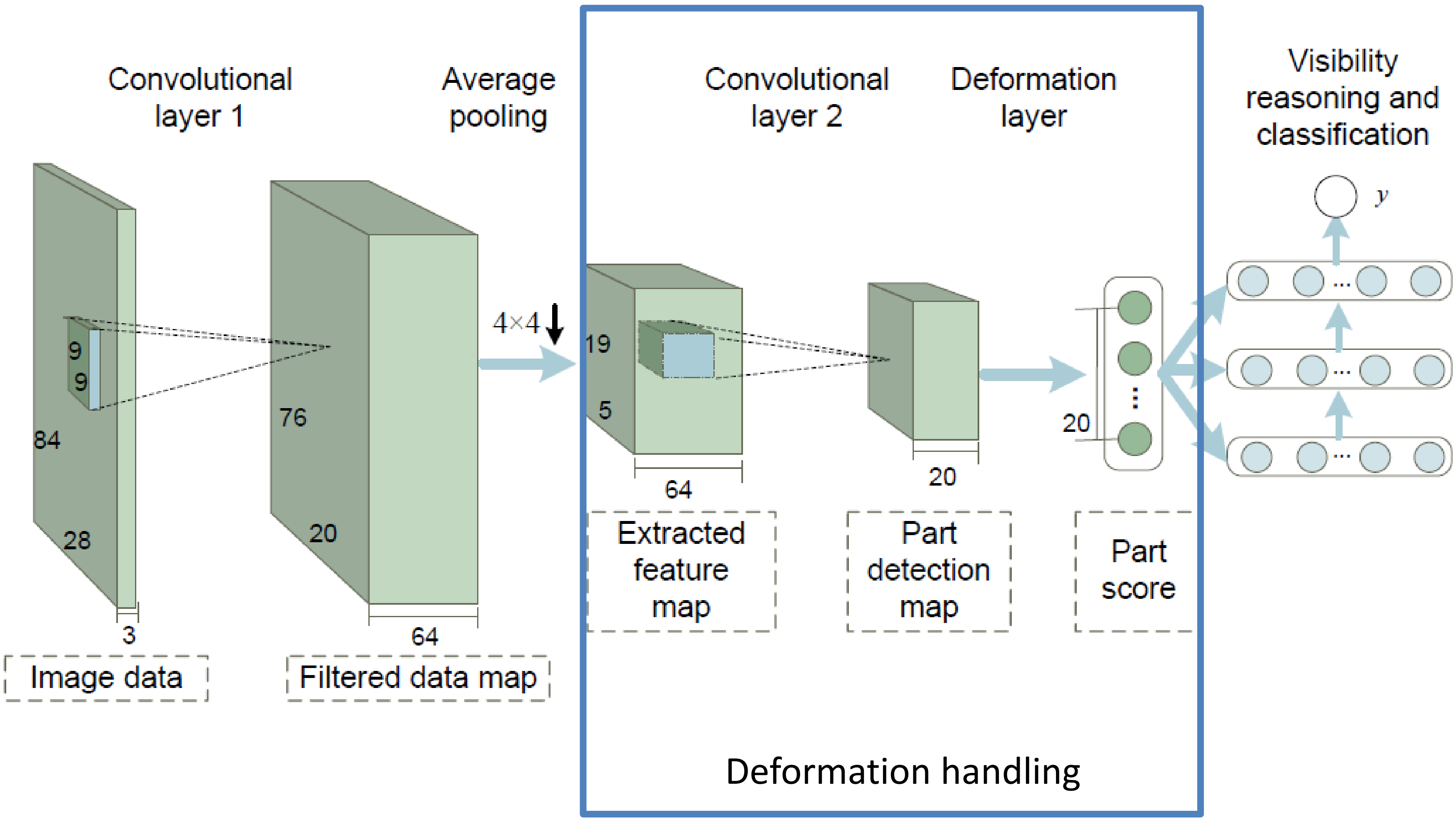


Part models

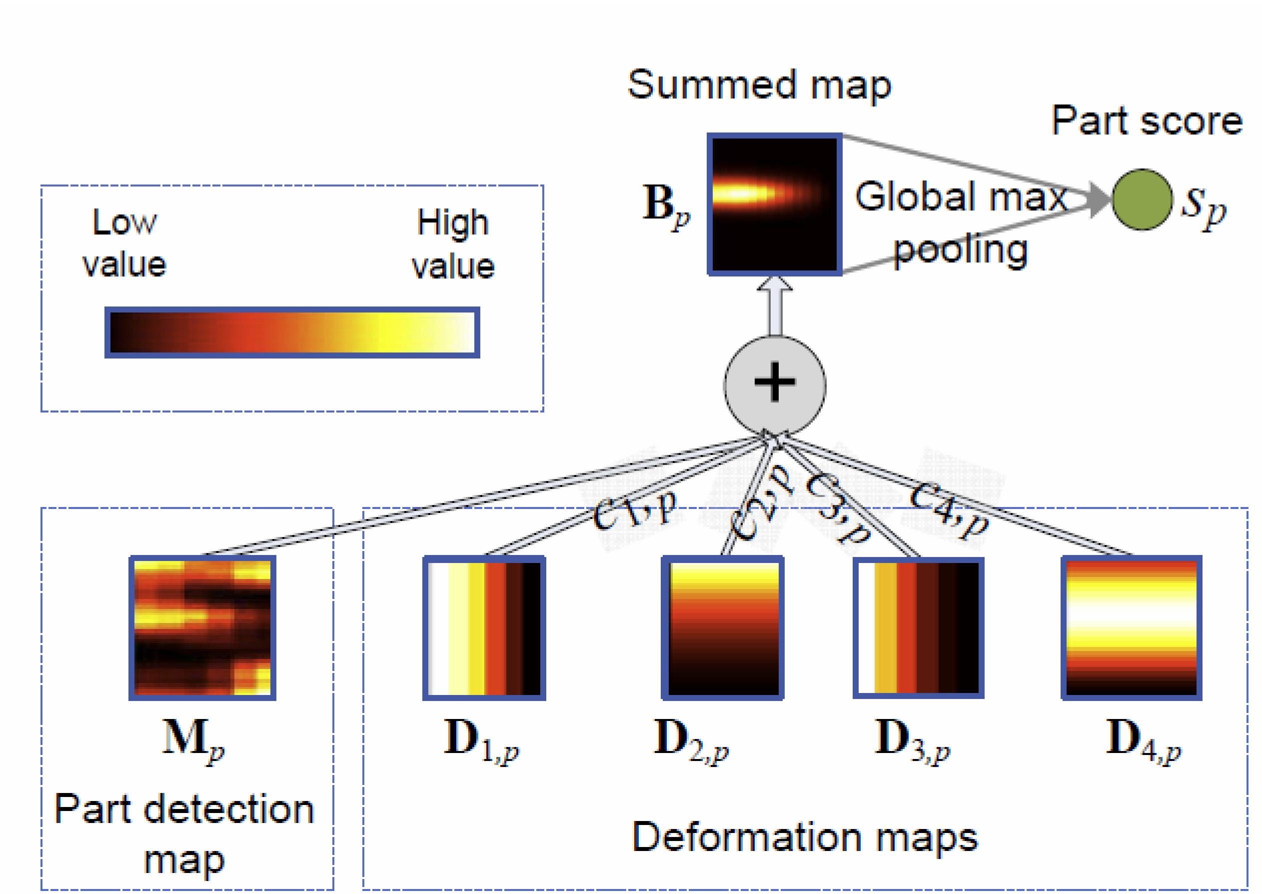


Learned filtered at the second convolutional layer

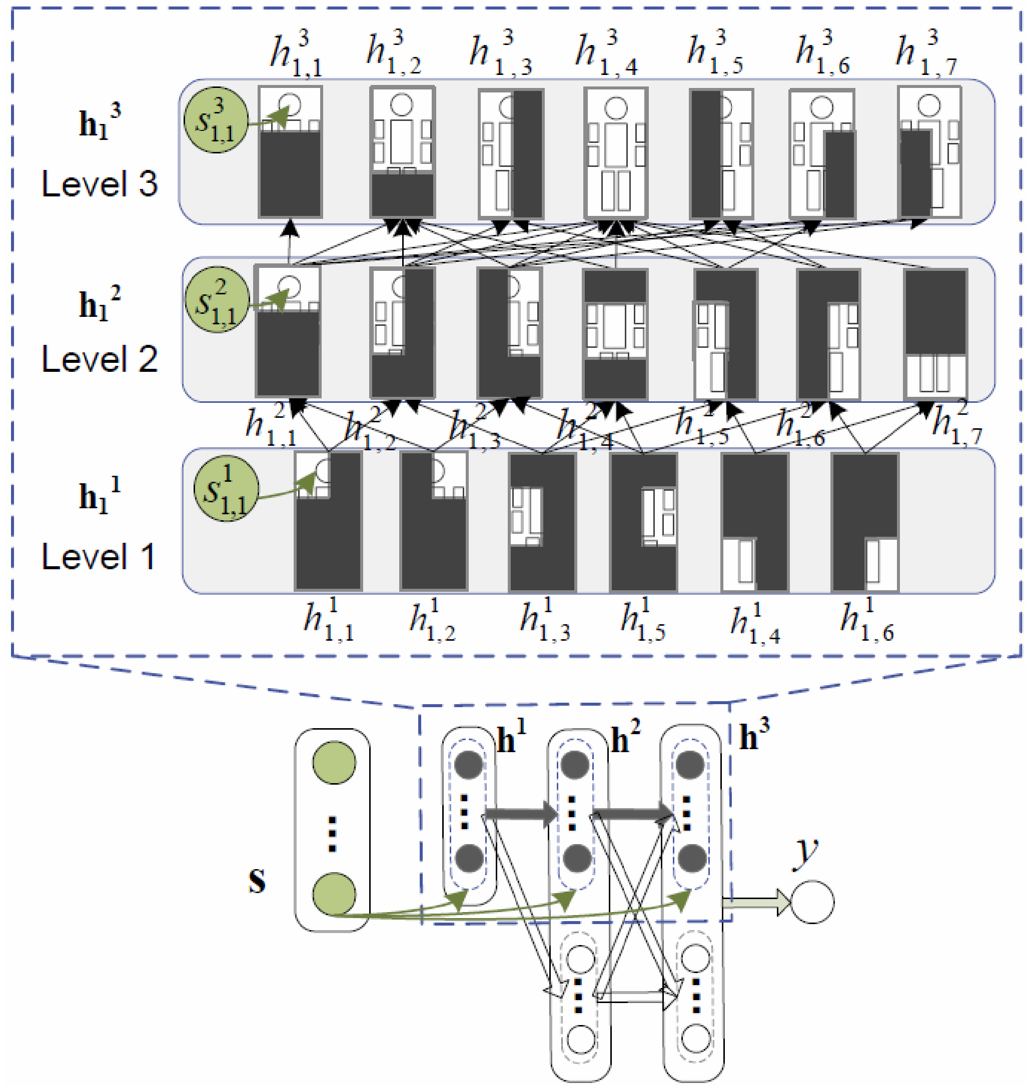
Our Joint Deep Learning Model

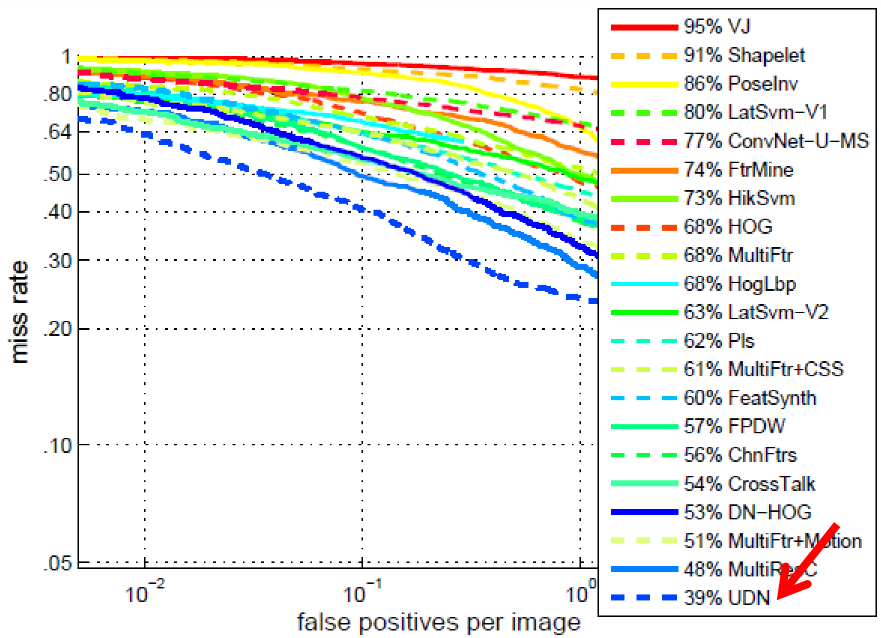


Deformation Layer

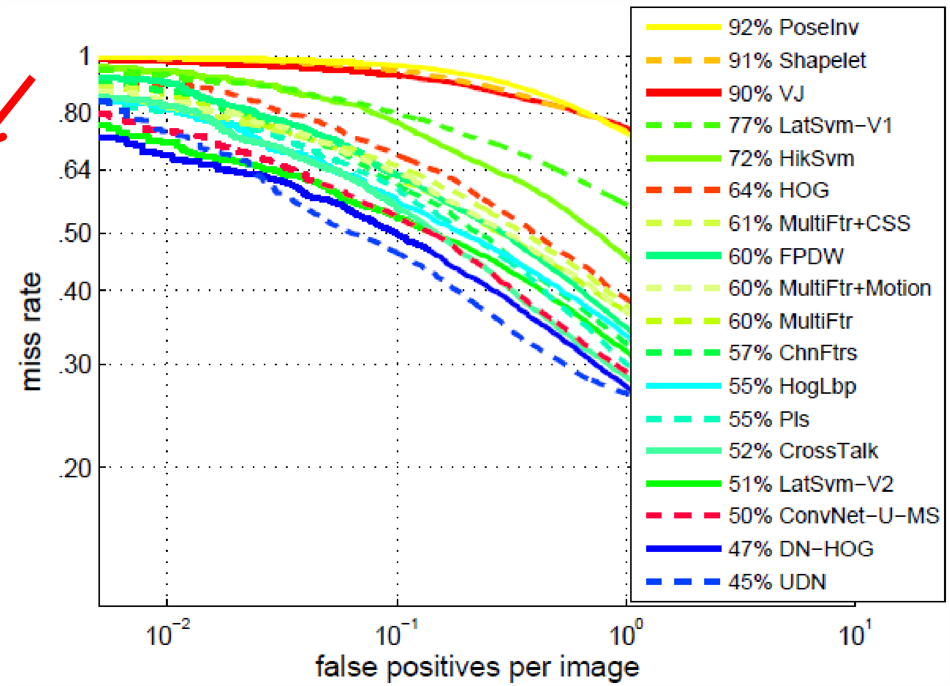


Visibility Reasoning with Deep Belief Net

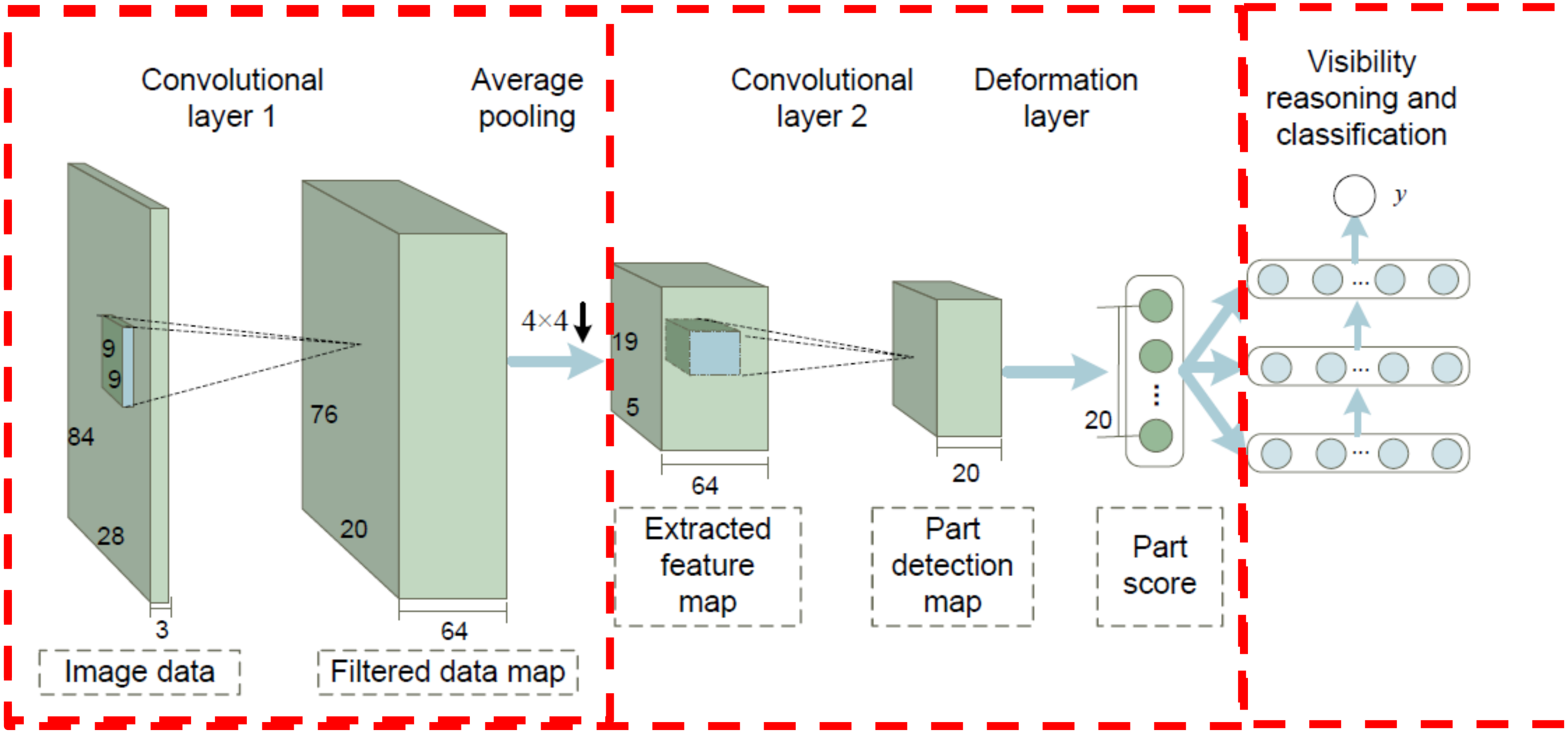


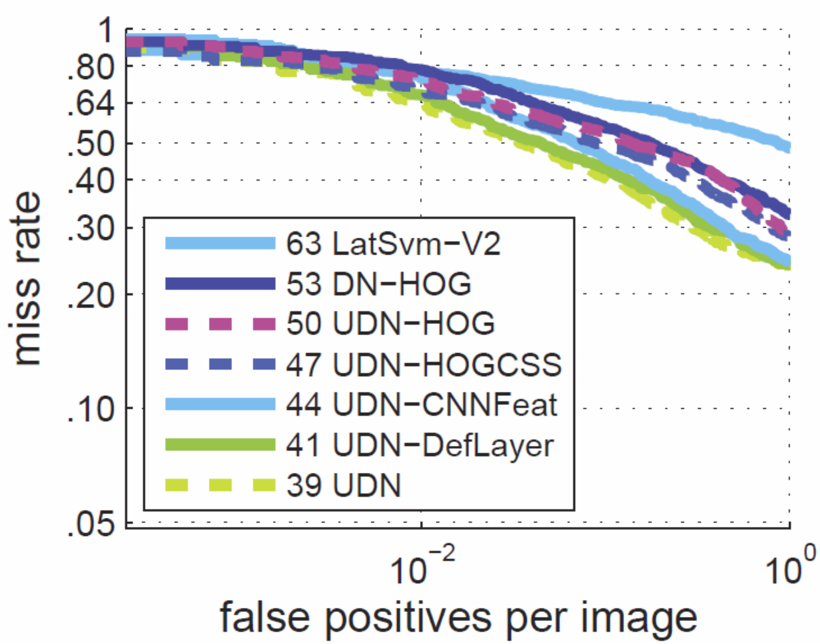


Results on Caltech Test



Results on ETHZ





DN-HOG
UDN-HOG
UDN-HOGCSS
UDN-CNNFeat
UDN-DefLayer

Multi-Stage Contextual Deep Learning:

✧ **Train different detectors for different types of samples**

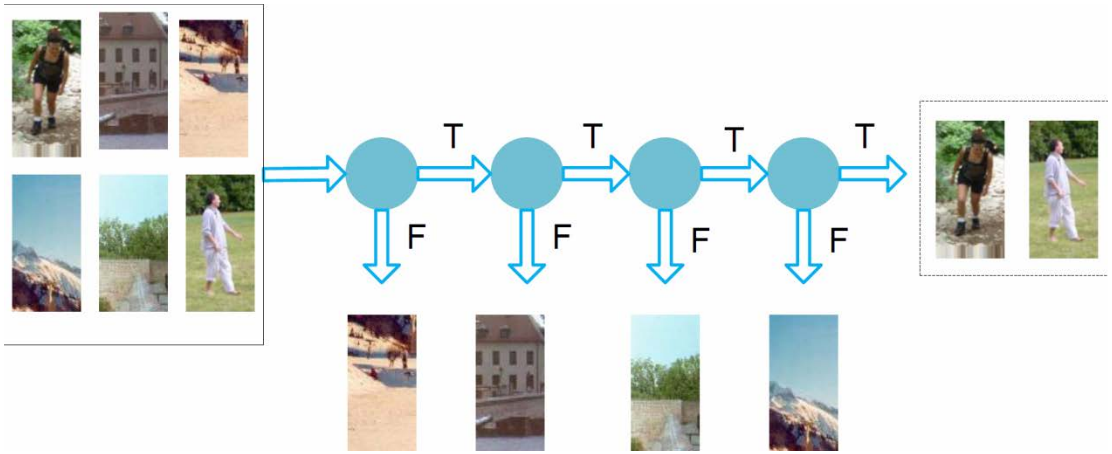
✧ **Model contextual information**

✧ **Stage-by-stage pretraining strategies**

X. Zeng, W. Ouyang and X. Wang, "Multi-Stage Contextual Deep Learning for Pedestrian Detection," ICCV 2013

Motivated by Cascaded Classifiers and Contextual Boost

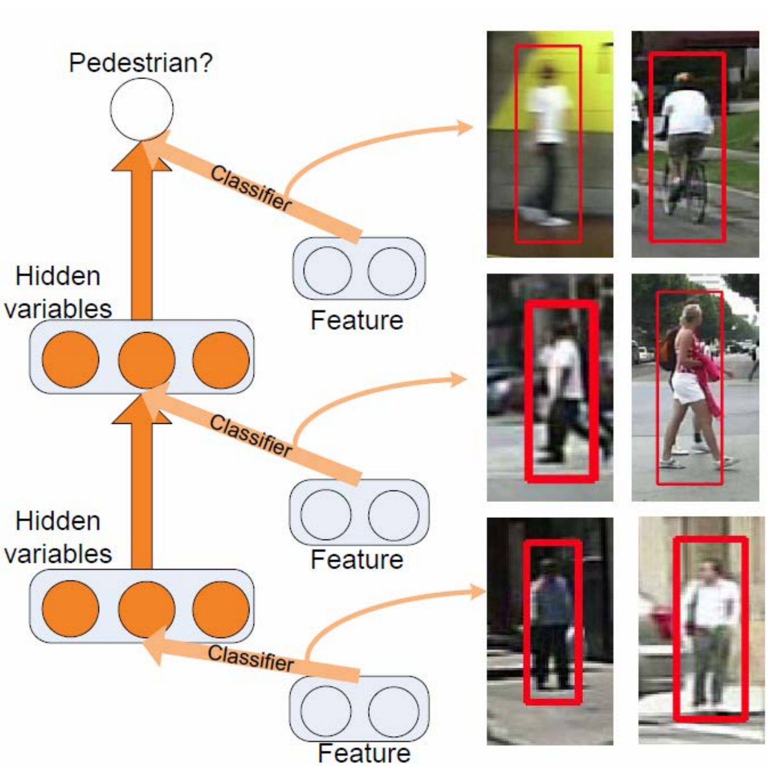
- The classifier of each stage deals with a specific set of samples
- The score map output by one classifier can serve as contextual information for the next classifier

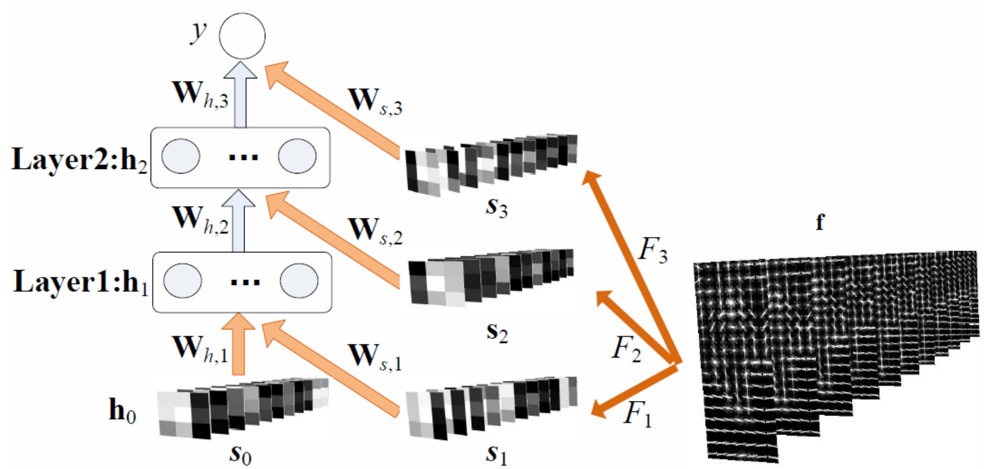


❖ Only pass one detection score to the next stage
❖ Classifiers are trained sequentially

Conventional cascaded classifiers for detection

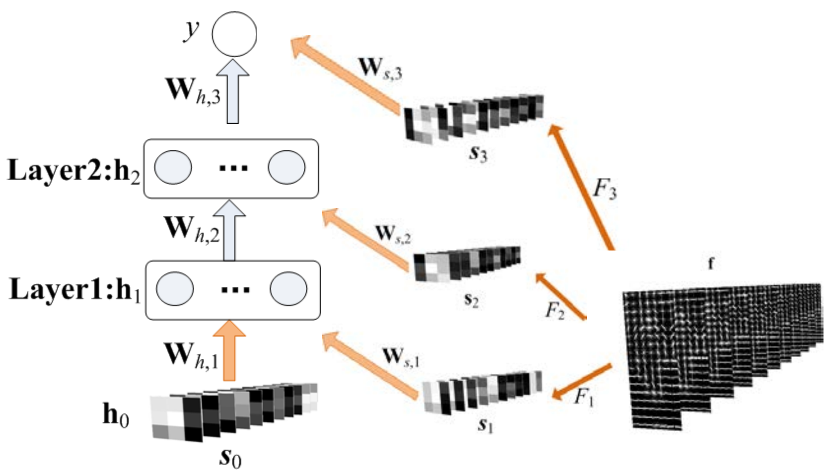
- Simulate the cascaded classifiers by mining hard samples to train the network stage-by-stage
- Cascaded classifiers are jointly optimized instead of being trained sequentially
- The deep model keeps the score map output by the current classifier and it serves as contextual information to support the decision at the next stage
- To avoid overfitting, a stage-wise pre-training scheme is proposed to regularize optimization





Training Strategies

- Unsupervised pre-train $\mathbf{W}_{h,i+1}$ layer-by-layer, setting $\mathbf{W}_{s,i+1} = 0$, $\mathbf{F}_{i+1} = 0$
- Fine-tune all the $\mathbf{W}_{h,i+1}$ with supervised BP
- Train $\mathbf{F}_{i+1}$ and $\mathbf{W}_{s,i+1}$ with BP stage-by-stage
- A correctly classified sample at the previous stage does not influence the update of parameters
- Stage-by-stage training can be considered as adding regularization constraints to parameters, i.e. some parameters are constrained to be zeros in the early training stages



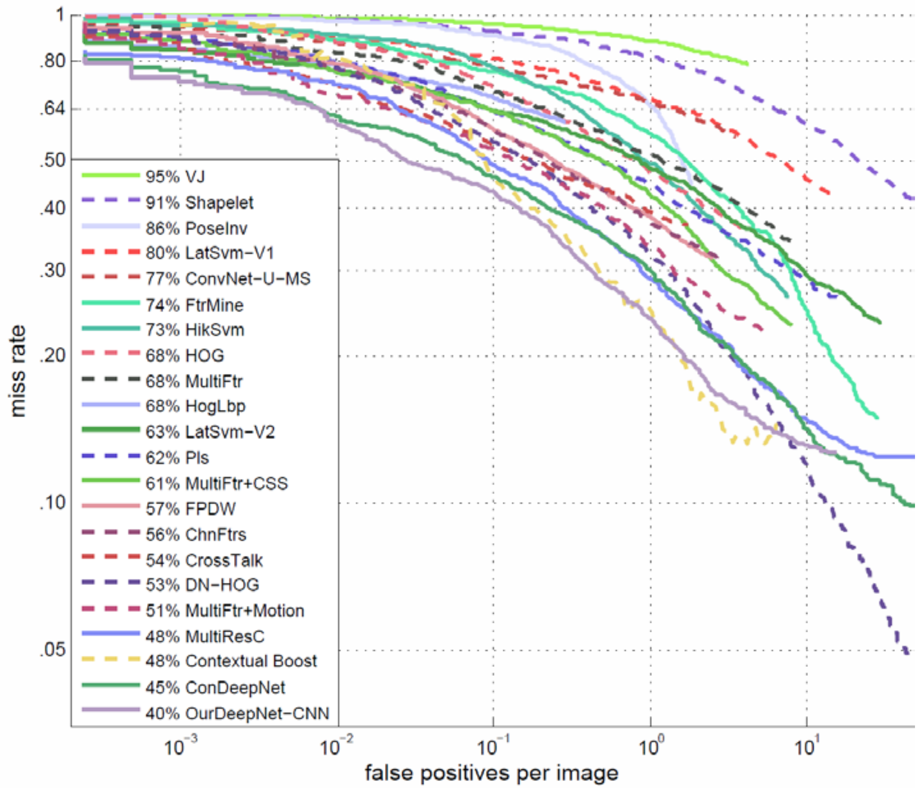
Log error function:

$$E = -l \log y - (1-l)\log(1-y)$$

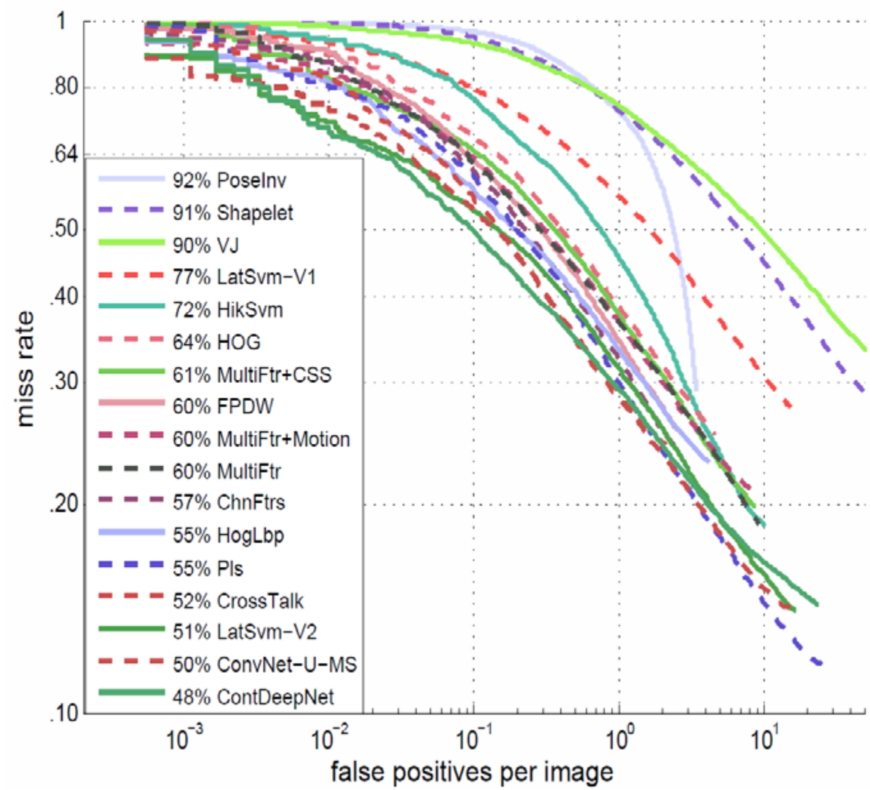
Gradients for updating parameters:

$$d\theta_{i,j} = -\frac{\partial E}{\partial \theta_{i,j}} = -\frac{\partial E}{\partial y}\frac{\partial y}{\partial \theta_{i,j}} = -(y-l)\frac{\partial y}{\partial \theta_{i,j}}$$

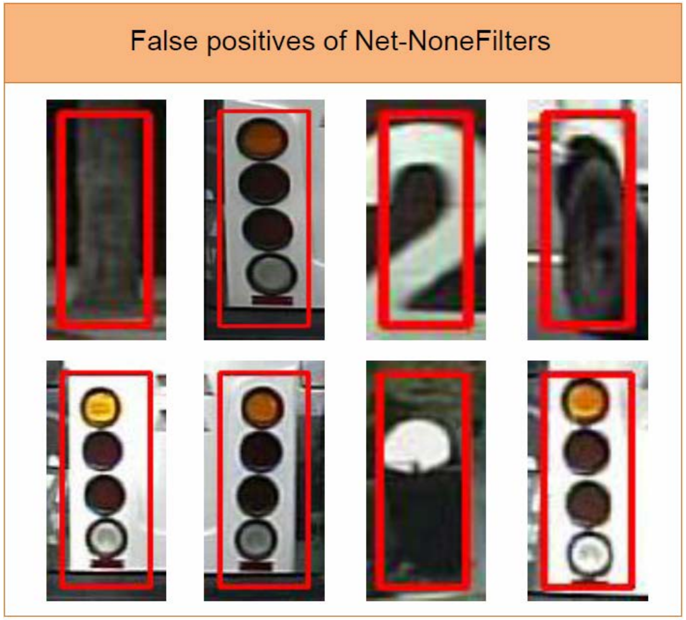
Experimental Results

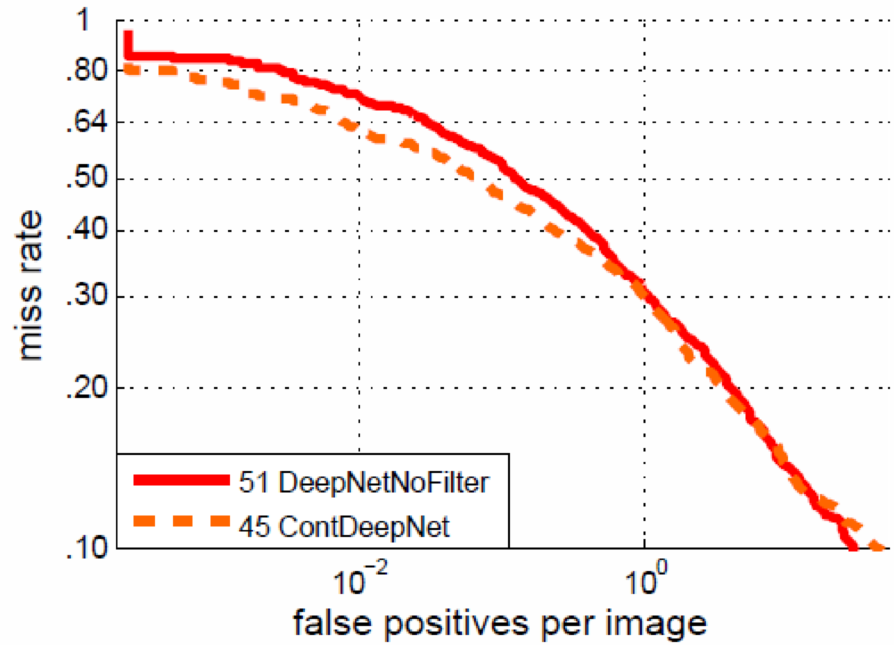


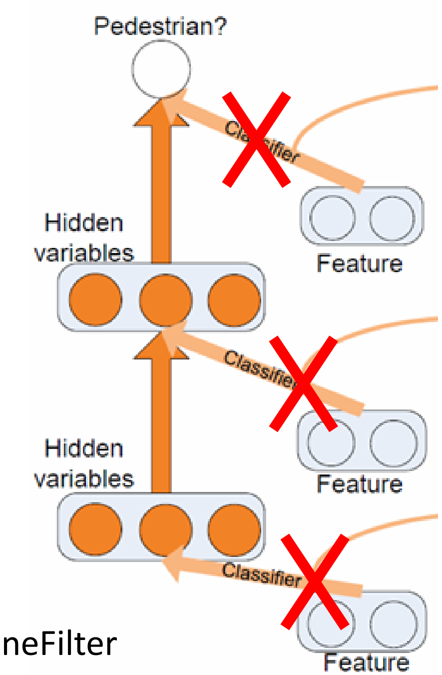
Caltech



ETHZ

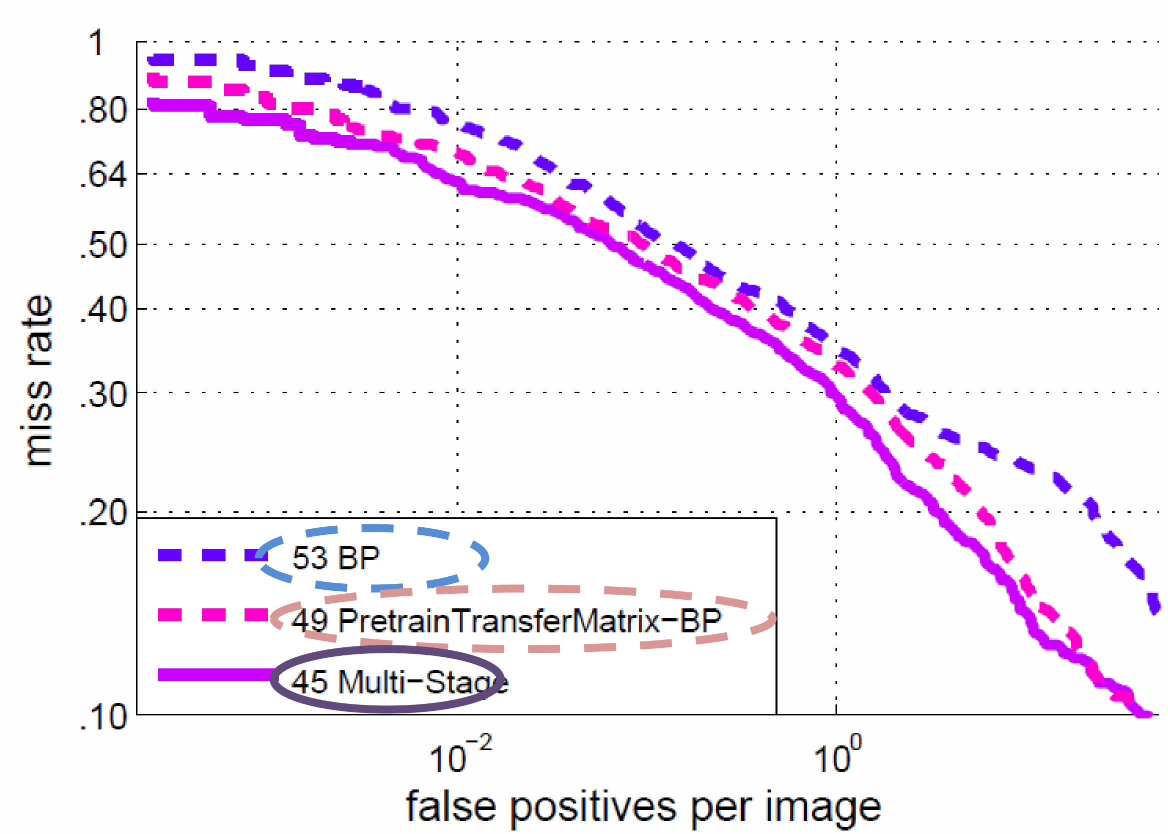






DeepNetNoneFilter

Comparison of Different Training Strategies



Network-BP: use back propagation to update all the parameters without pre-training
PretrainTransferMatrix-BP: the transfer matrices are unsupervised pertrained, and then all the parameters are fine-tuned
Multi-stage: our multi-stage training strategy

Switchable Deep Network

- ✧ **Use mixture components to model complex variations of body parts**
- ✧ **Use salience maps to depress background clutters**
- ✧ **Help detection with segmentation information**

P. Luo, Y. Tian, X. Wang, and X. Tang, "Switchable Deep Network for Pedestrian Detection", CVPR 2014

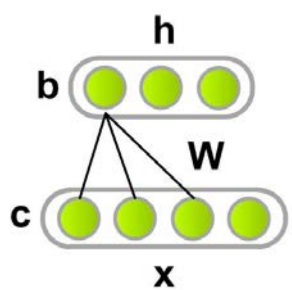
Switchable Deep Network for Pedestrian Detection

- *Background clutter* and large variations of pedestrian appearance.
- **Proposed Solution.** A Switchable Deep Network (SDN) for learning the foreground map and removing the effect background clutter.

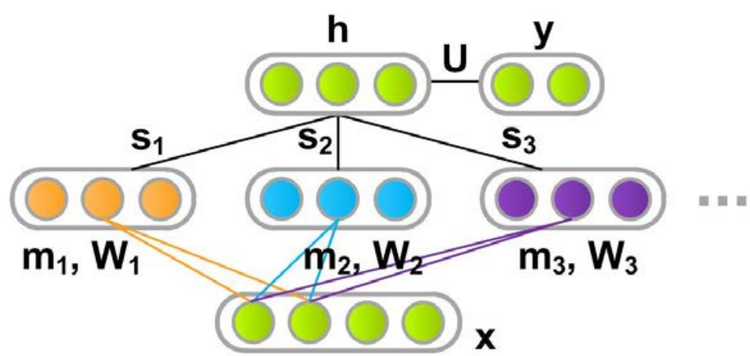
Switchable Deep Network for Pedestrian Detection

- Switchable Restricted Boltzmann Machine

$$E(\mathbf{x},\mathbf{y},\mathbf{h},\mathbf{s},\mathbf{m};\Theta) = -\sum_{k=1}^{K} s_k \mathbf{h}_k^T(\mathbf{W}_k(\mathbf{x}\circ\mathbf{m}_k)+\mathbf{b}_k) - \sum_{k=1}^{K} s_k \mathbf{c}_k^T(\mathbf{x}\circ\mathbf{m}_k) - \mathbf{y}^T\mathbf{U}\sum_{k=1}^{K} s_k \mathbf{h}_k - \mathbf{d}^T\mathbf{y},$$



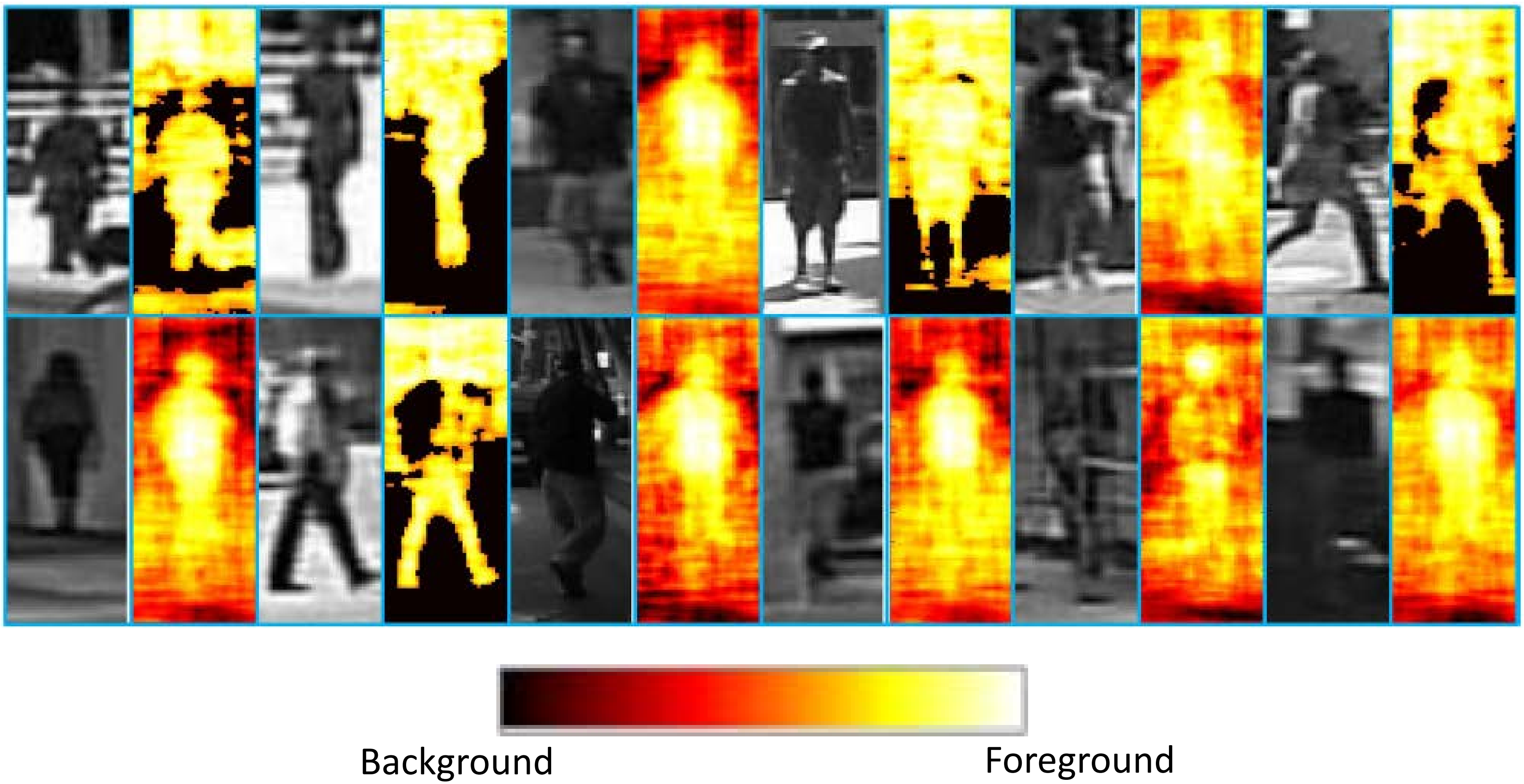
(a) RBM



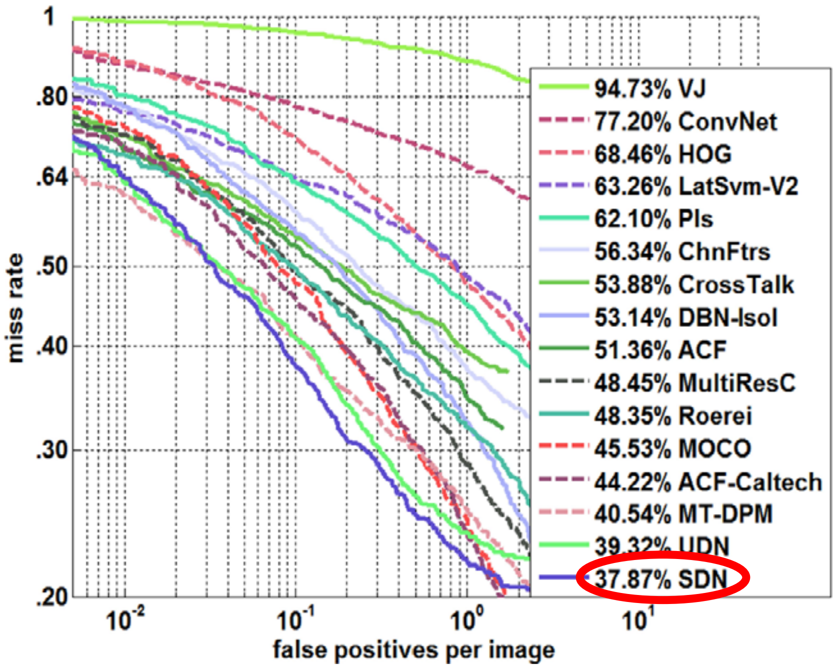
(b) Switchable RBM

Switchable Deep Network for Pedestrian Detection

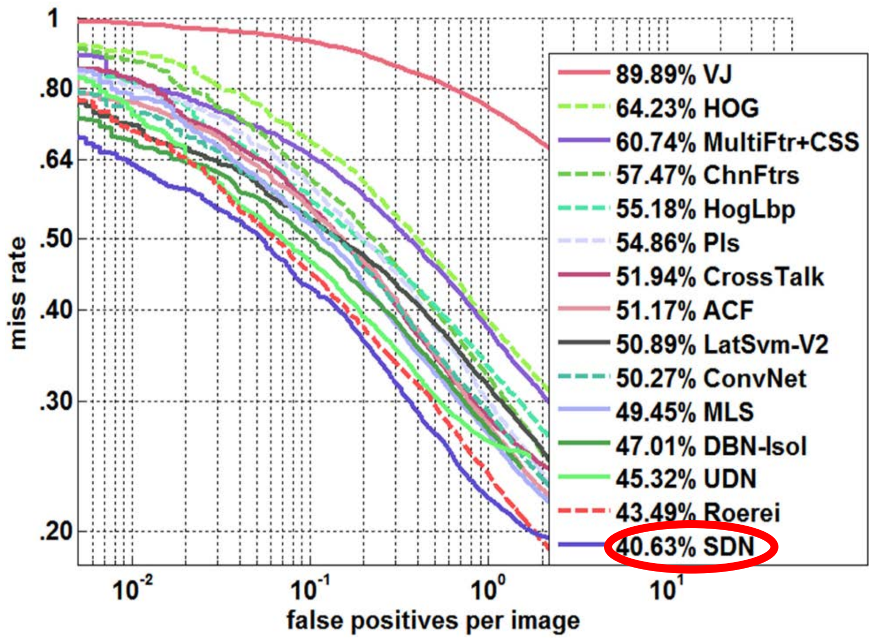
- Switchable Restricted Boltzmann Machine



Switchable Deep Network for Pedestrian Detection



(a) Performance on Caltech Test



(b) Performance on ETH

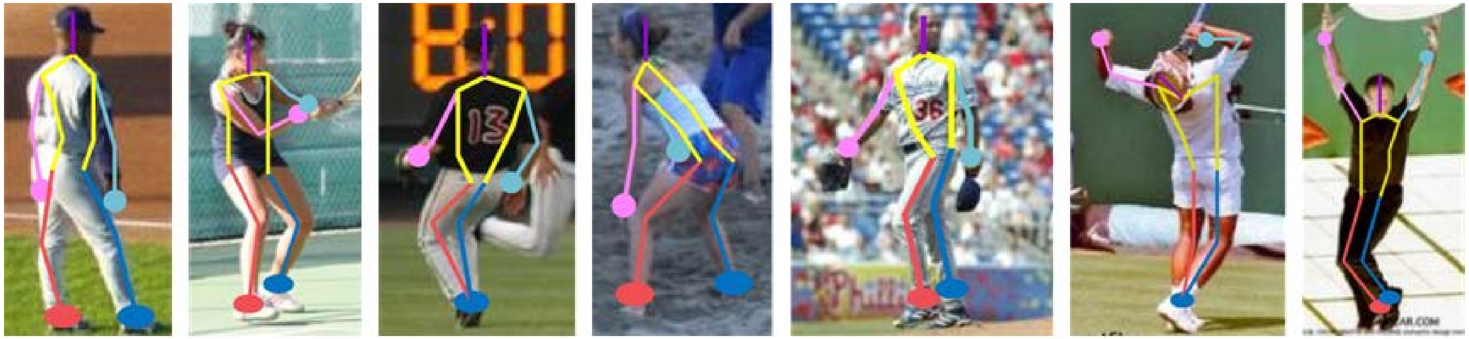
Human Part Localization

- Contextual information is important to segmentation as well as detection

Human part localization

- Facial Keypoint Detection
- Human pose estimation

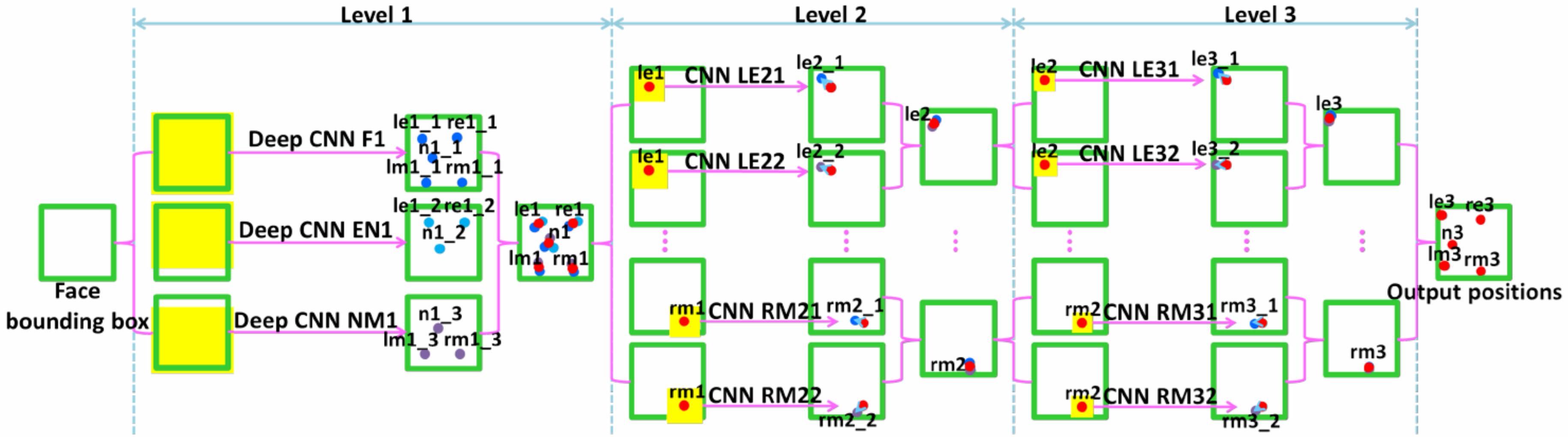
Sun et al. CVPR' 13



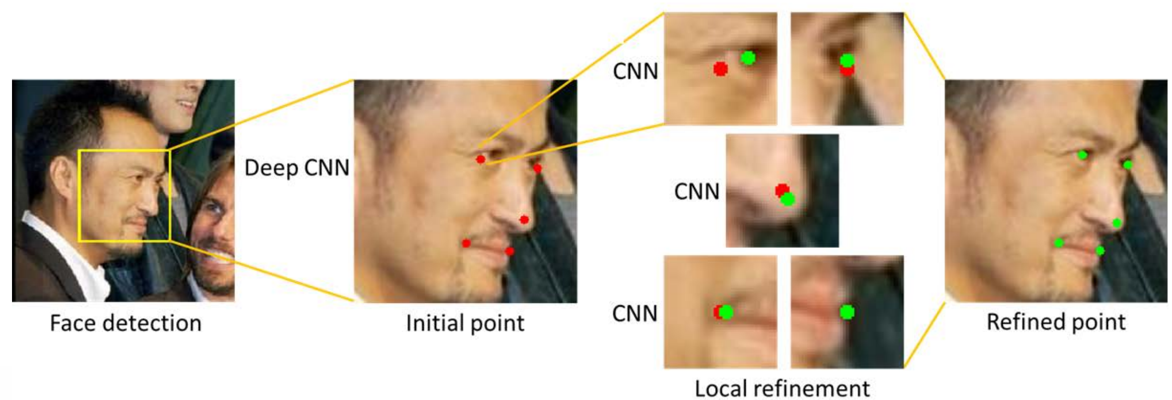
Ouyang et al. CVPR' 14

Facial Keypoint Detection

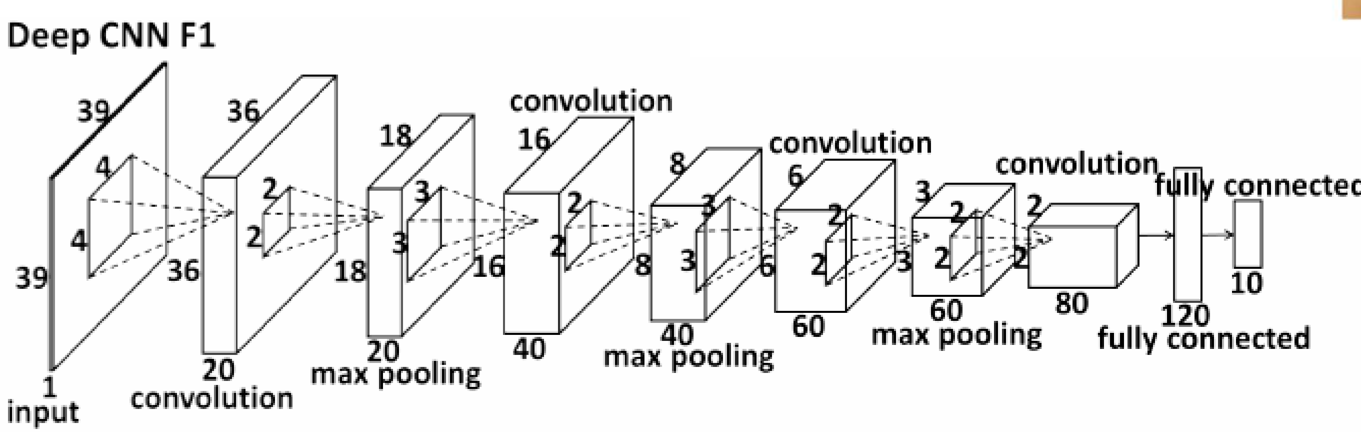
- Y. Sun, X. Wang and X. Tang, “Deep Convolutional Network Cascade for Facial Point Detection,” CVPR 2013



Level 1
Level 2
Level 3
Face
bounding box
Deep CNN F1
Deep CNN EN1
Deep CNN NM1
le1_1 re1_1
n1_1
lm1_1 rm1_1
le1_2 re1_2
n1_2
n1_3
lm1_3 rm1_3
le1 re1
n1
lm1 rm1
le1
CNN LE21
le1
CNN LE22
rm1
CNN RM21
rm1
CNN RM22
le2_1
le2_2
rm2_1
rm2_2
le2
rm2
le2
CNN LE31
le2
CNN LE32
rm2
CNN RM31
rm2
CNN RM32
le3_1
le3_2
rm3_1
rm3_2
le3
rm3
le3 re3
n3
lm3 rm3
Output positions



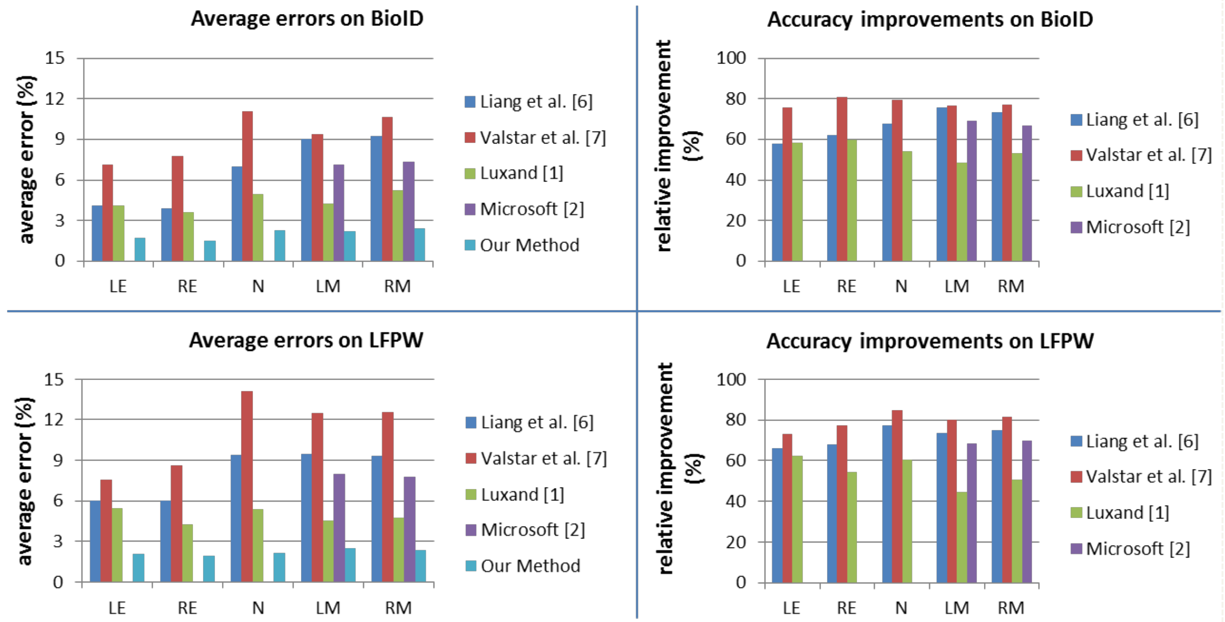
Deep CNN
CNN
CNN
CNN
Face detection
Initial point
Local refinement
Refined point



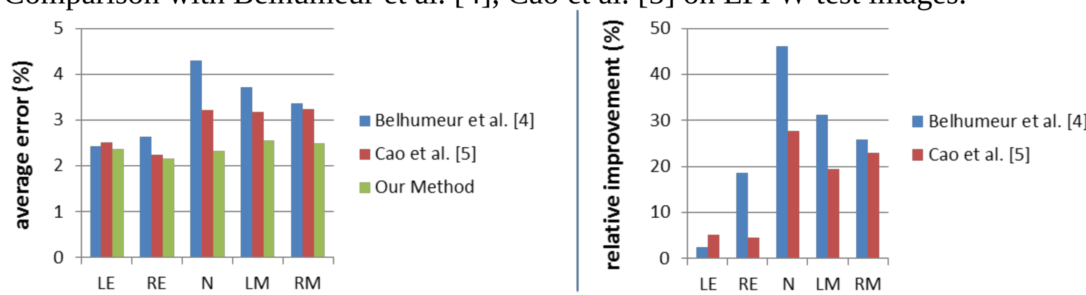
Deep CNN F1
39
39
4
4
1
input
36
36
2
2
20
convolution
18
18
3
3
20
max pooling
convolution
16
16
2
2
40
8
8
3
3
40
max pooling
convolution
6
6
2
2
60
3
3
2
2
60
max pooling
convolution
2
2
80
fully connected
120
fully connected
10

Comparison with Liang et al. [6], Valstar et al. [7], Luxand Face SDK [1] and Microsoft Research Face SDK [2] on BioID and LFPW.

$$\text{Relative improvement} = \frac{\text{reduced average error}}{\text{average error of the method in comparison}}.$$



Comparison with Belhumeur et al. [4], Cao et al. [5] on LFPW test images.



1. http://www.luxand.com/facesdk/
2. http://research.microsoft.com/en-us/projects/facesdk/.
3. O. Jesorsky, K. J. Kirchberg, and R. Frischholz. Robust face detection using the hausdorff distance. In Proc. AVBPA, 2001.
4. P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. In Proc. CVPR, 2011.
5. X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. In Proc. CVPR, 2012.
6. L. Liang, R. Xiao, F. Wen, and J. Sun. Face alignment via component-based discriminative search. In Proc. ECCV, 2008.
7. M. Valstar, B. Martinez, X. Binefa, and M. Pantic. Facial point detection using boosted regression and graph models. In Proc. CVPR, 2010.

Validation.

BioID.

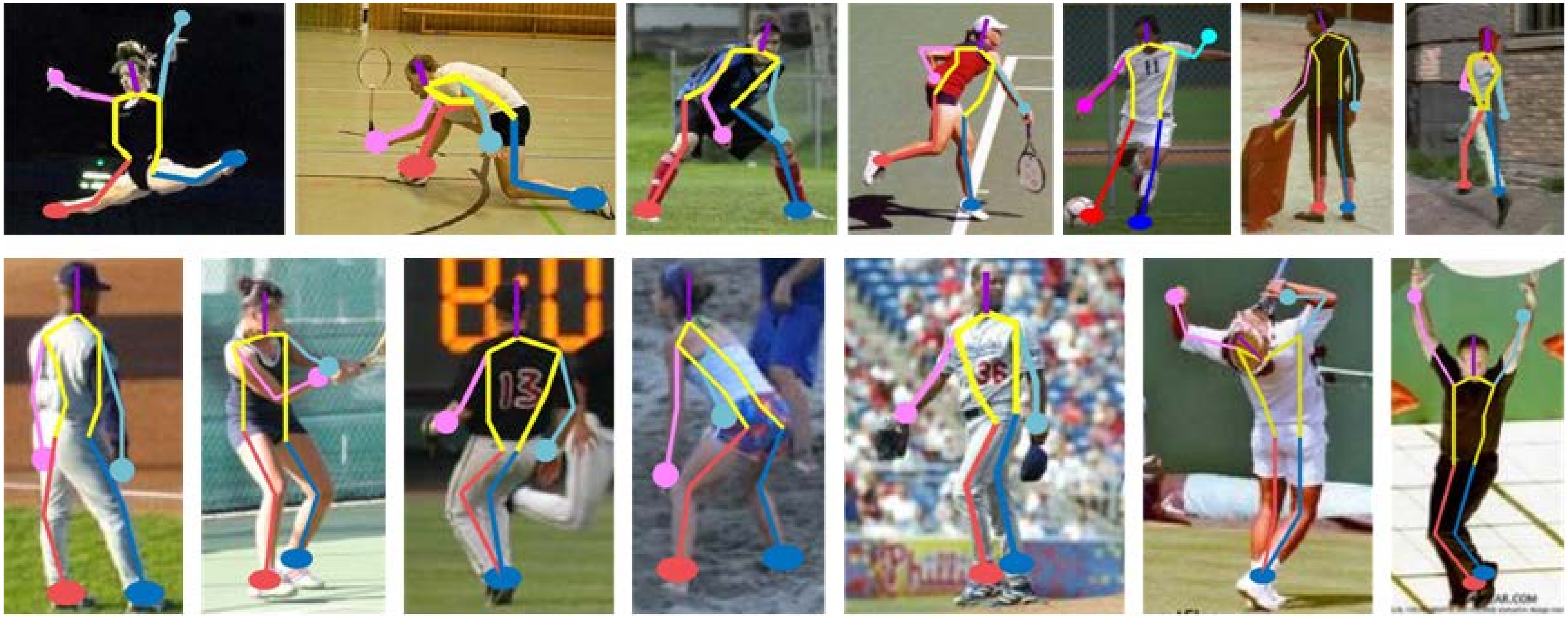
LFPW.

Benefits of Using Deep Model

- The first network that takes the whole face as input needs **deep** structures to extract **high-level** features
- Take the full face as input to make full use of texture context information over the entire face to locate each keypoint
- Since the networks are trained to predict all the keypoints simultaneously, the geometric constraints among keypoints are implicitly encoded

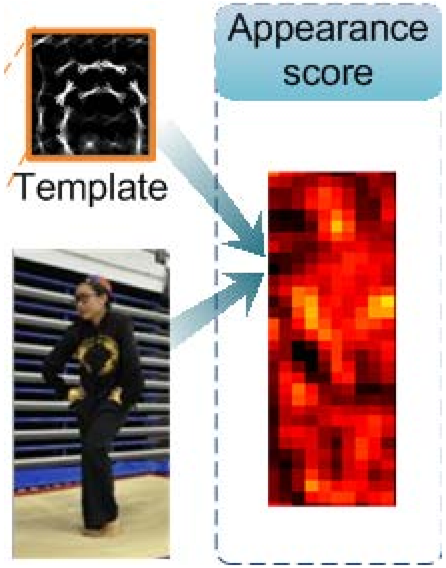
Human pose estimation

- W. Ouyang, X. Chu and X. Wang, “Multi-source Deep Learning for Human Pose Estimation” CVPR 2014.



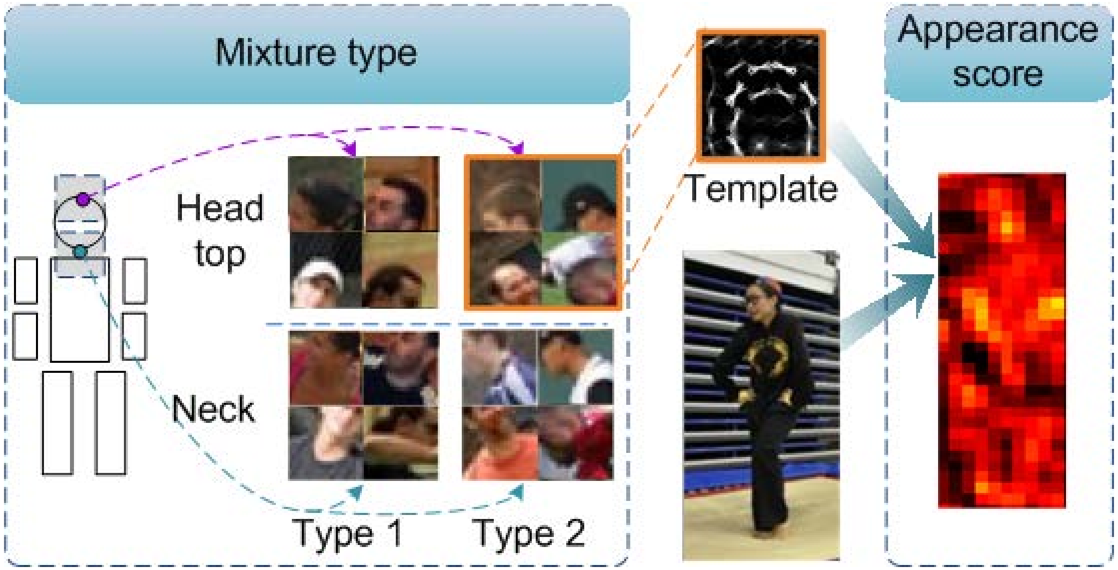
Multiple information sources

- Appearance



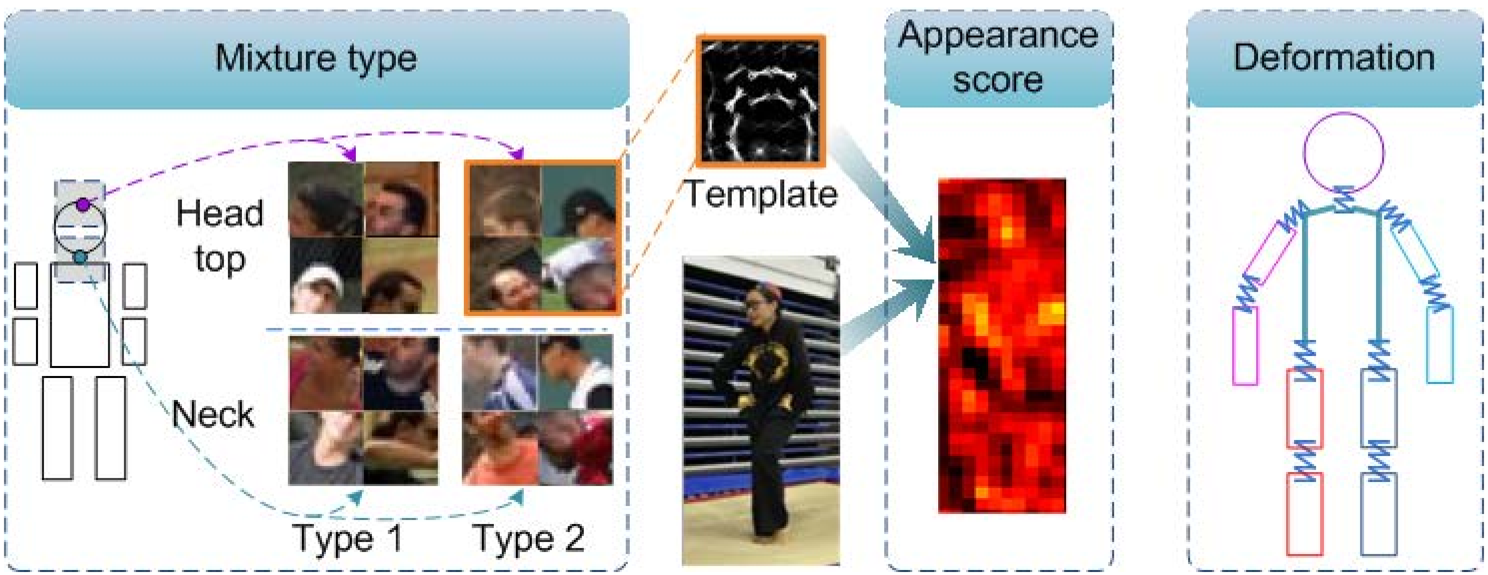
Multiple information sources

- Appearance
- Appearance mixture type

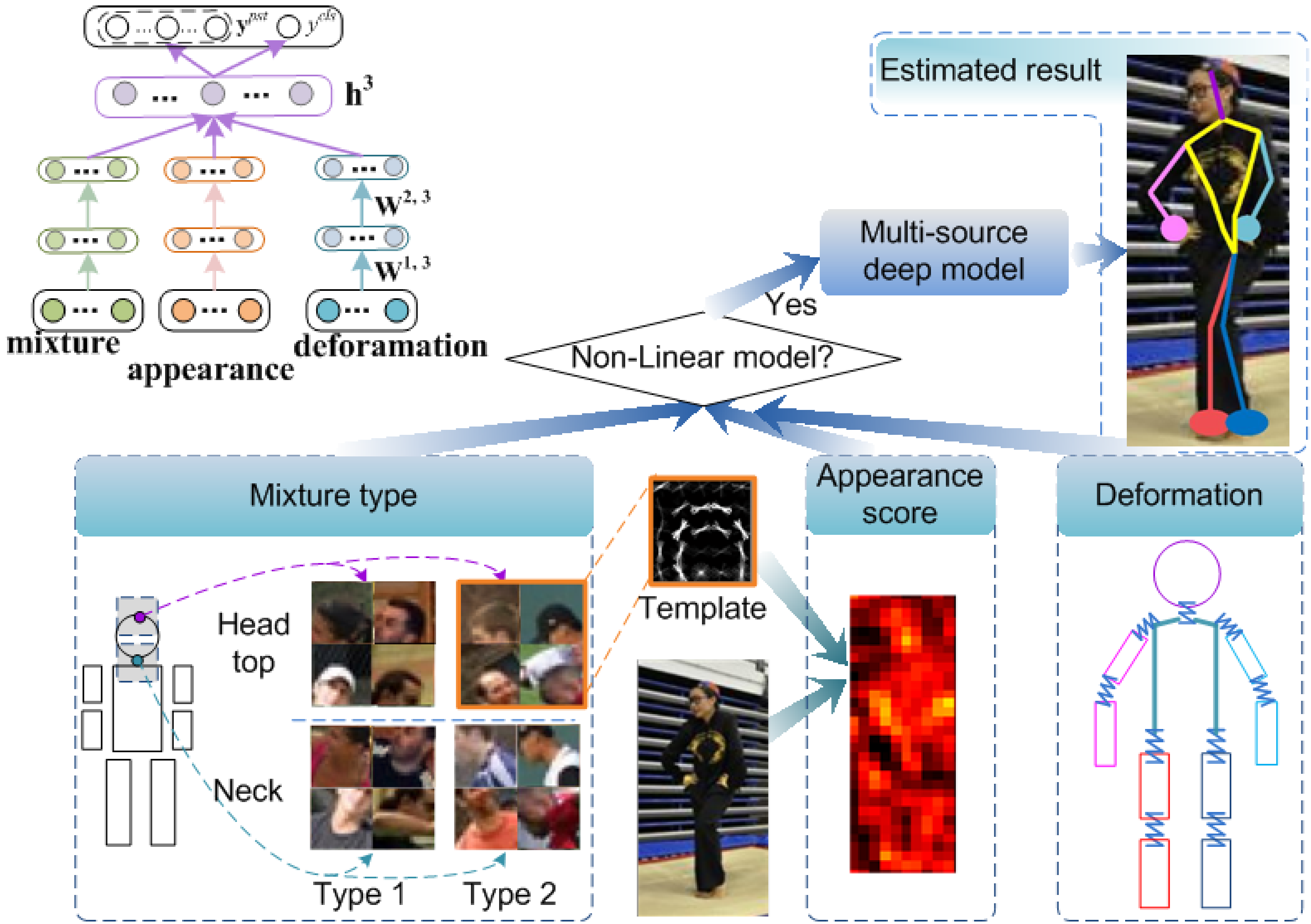


Multiple information sources

- Appearance
- Appearance mixture type
- Deformation



Multi-source deep model



Experimental results

PARSE							
Method	**Torso**	**U.leg**	**L.leg**	**U.arm**	**L.arm**	**head**	**Total**
Yang&Ramanan CVPR'11	82.9	68.8	60.5	63.4	42.4	82.4	63.6
Multi-source deep learning	89.3	78.0	72.0	67.8	47.8	89.3	71.0

UIUC People							
Method	**Torso**	**U.leg**	**L.leg**	**U.arm**	**L.arm**	**head**	**Total**
Yang&Ramanan CVPR'11	81.8	65.0	55.1	46.8	37.7	79.8	57.0
Multi-source deep learning	89.1	72.9	62.4	56.3	47.6	89.1	65.6

LSP							
Method	**Torso**	**U.leg**	**L.leg**	**U.arm**	**L.arm**	**head**	**Total**
Yang&Ramanan CVPR'11	82.9	70.3	67.0	56.0	39.8	79.3	62.8
Multi-source deep learning	85.8	76.5	72.2	63.3	46.6	83.1	68.6

Up to 8.6 percent accuracy improvement with global geometric constraints

Experimental results

Left: mixtire-of-parts (Yang&Ramanan CVPR'11)
Right: Multi-source deep learning

General Object Detection

- ✧ Pretraining
- ✧ Model deformation of object parts, which are shared across classes
- ✧ Contextual modeling

Object detection

Pascal VOC

~ 20 object classes

Training: ~ 5,700 images

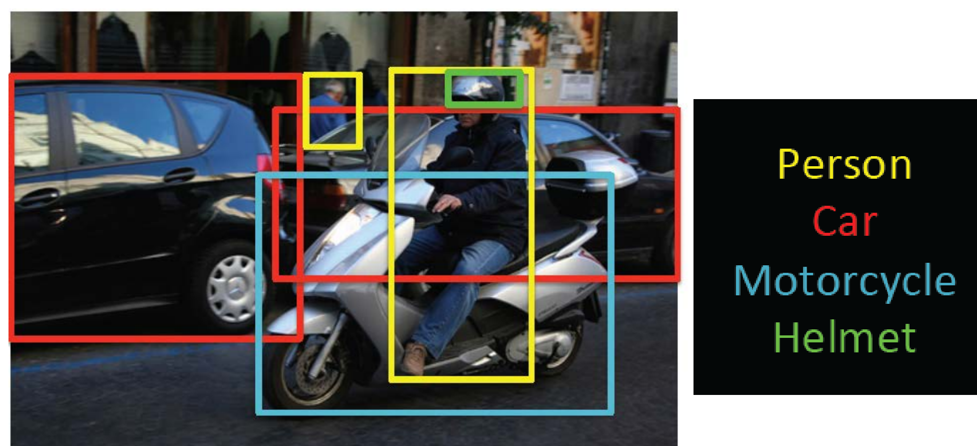
Testing: ~10,000 images

Image-net ILSVRC

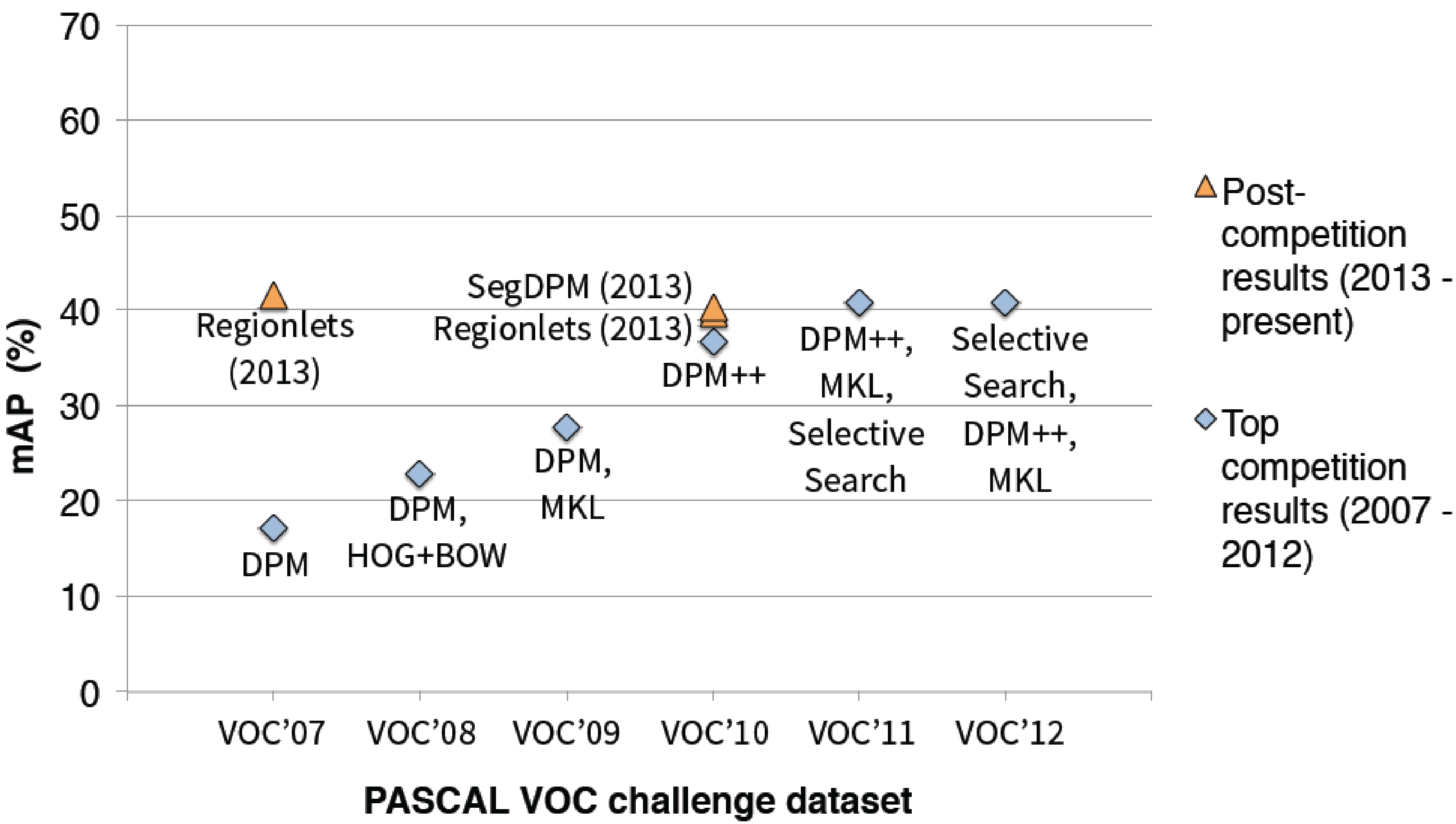
~ 200 object classes

Training: ~ 395,000 images

Testing: ~ 40,000 images

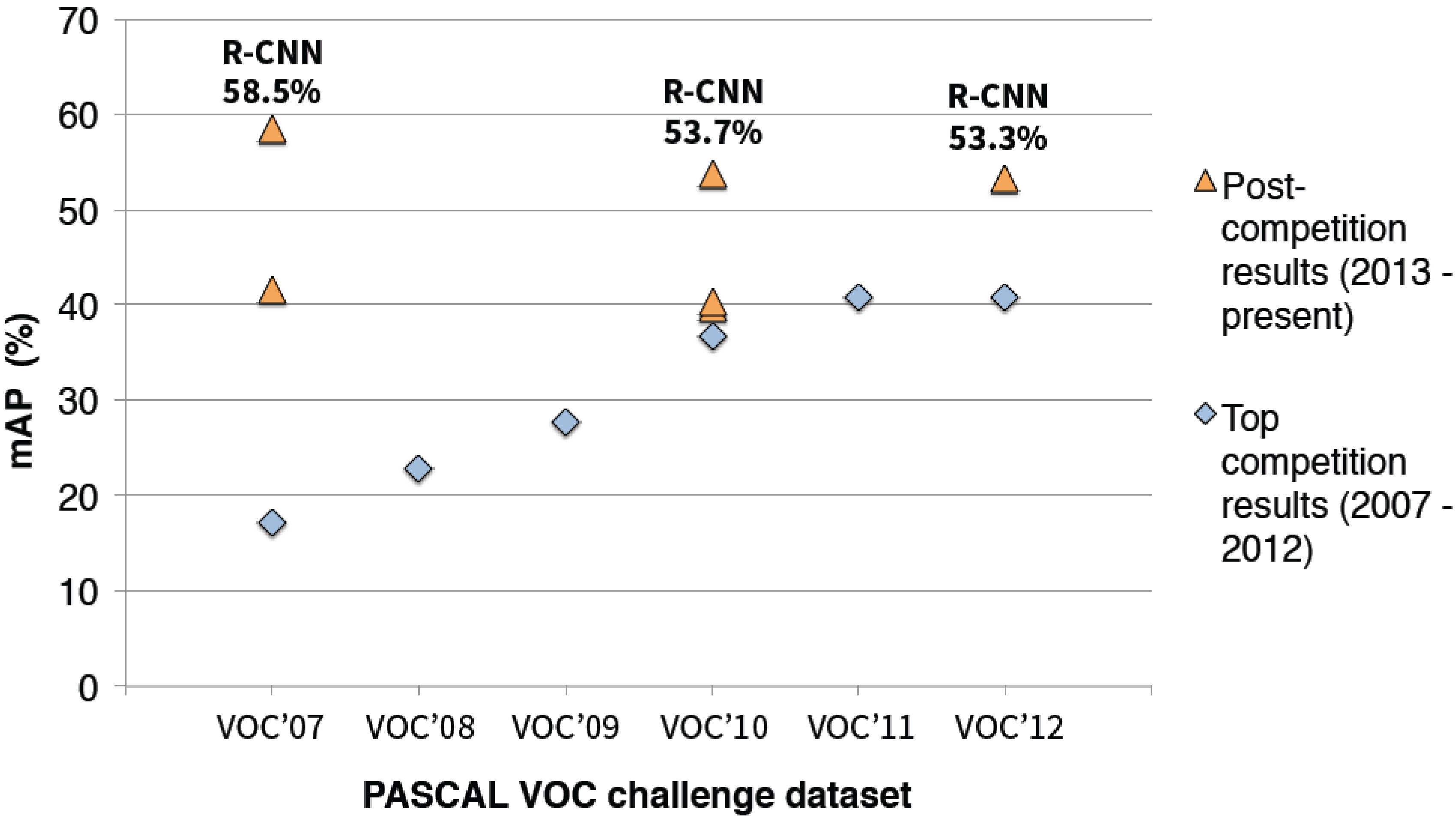


SIFT, HOG, LBP, DPM …



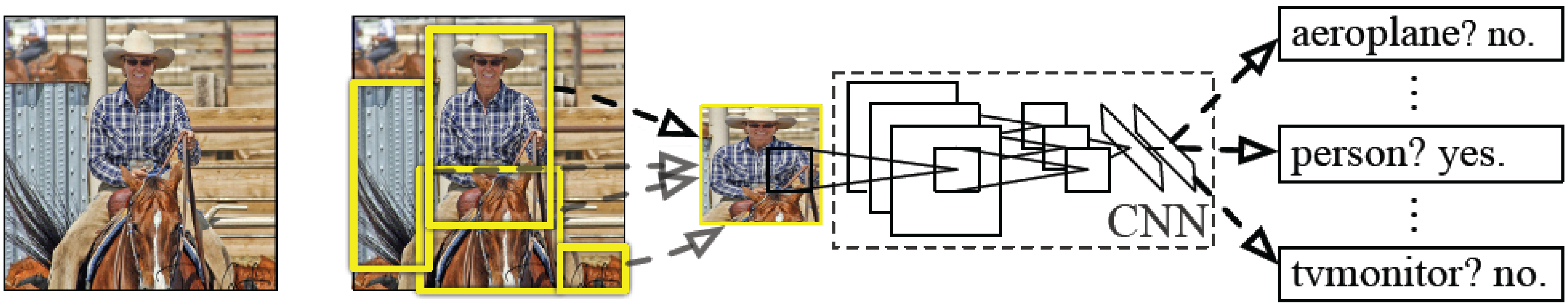
[Regionlets. Wang et al. ICCV'13] [SegDPM. Fidler et al. CVPR'13]

With CNN features



R. Girshick, J. Donahue, T. Darrell, J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," CVPR, 2014.

R-CNN: regions + CNN features



Input image

Extract region proposals (~2k/image)

Compute CNN features

2-class linear SVM

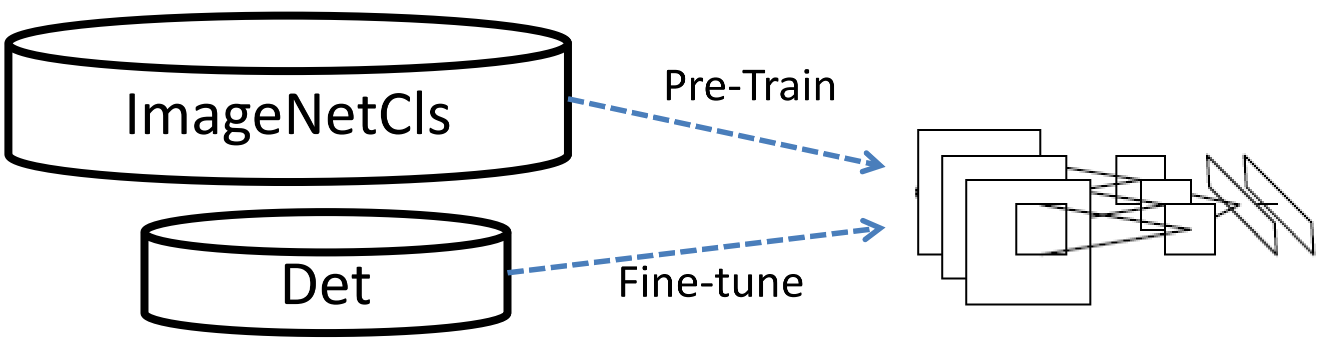
Region:
91.6%/98% recall rate on ImageNet/PASCAL
Selective Search [van de Sande, Uijlings et al. IJCV 2013].

Deep model from Krizhevsky, Sutskever & Hinton. NIPS 2012

SVM: Liblinear

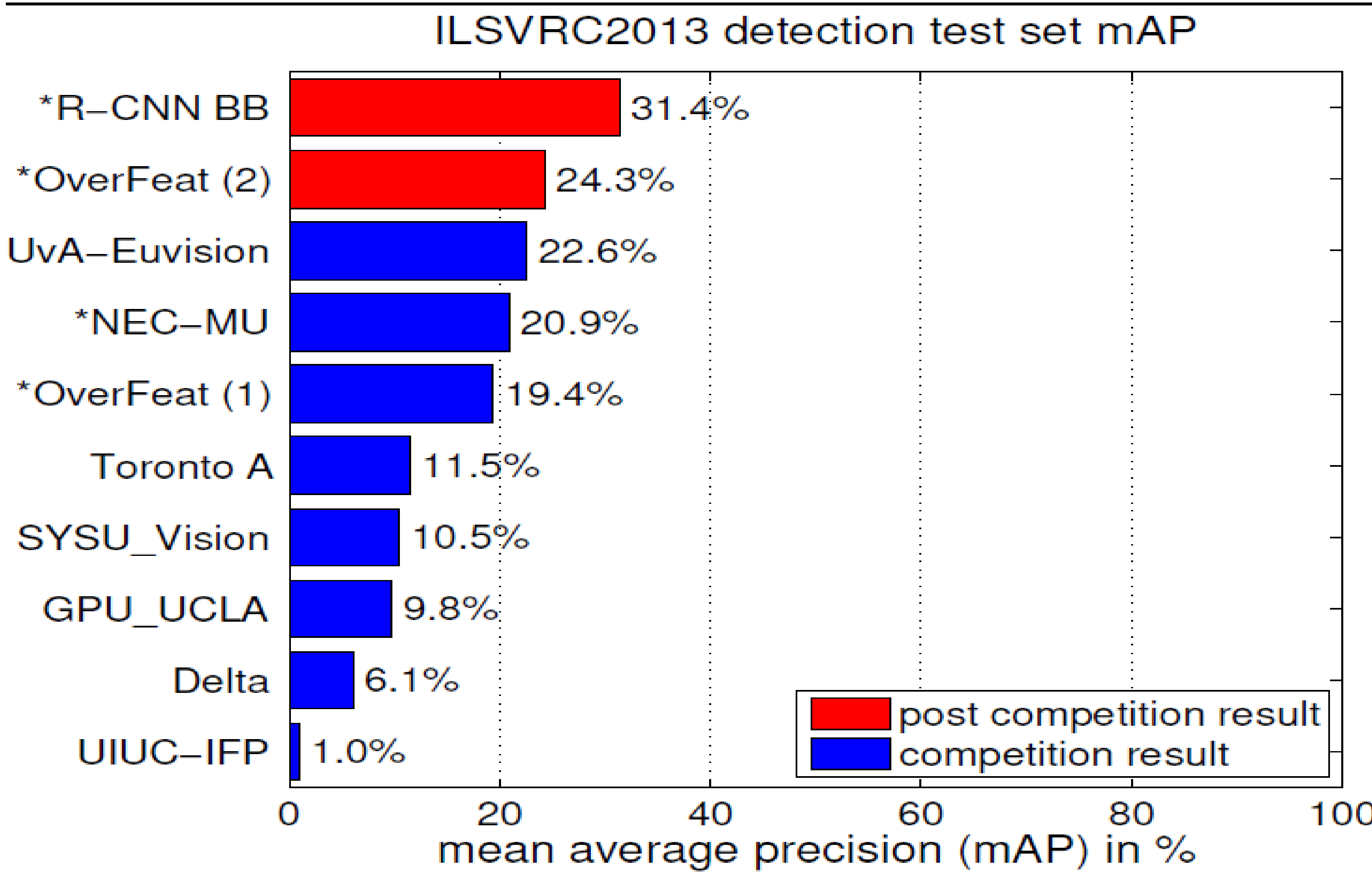
RCNN: deep model training

- Pretrain for the 1000-way ILSVRC image classification task (1.2 million images)
- Fine-tune the CNN for detection
  - ➢ Transfer the representation learned from ILSVRC Classification to PASCAL (or ImageNet) detection



Network from Krizhevsky, Sutskever & Hinton. NIPS 2012
Also called “AlexNet”

Experimental results on ILSVRC 2013



Experimental results on ILSVRC 2014

	GoogLeNet (Google)	DeepID-Net (CUHK)	DeepInsight	UvA-Euvision	Berkley Vision	RCNN
Model average	0.439	**0.439**	0.405	n/a	n/a	n/a
Single model	0.380	**0.427**	0.402	0.354	0.345	0.314

DeepID-Net: deformable **deep** convolutional neural networks for gener**i**c object **d**etection

W. Ouyang and X. Wang, et al. “DeepID-Net: deformable deep convolutional neural networks for object detection,” CVPR 2015.

RCNN

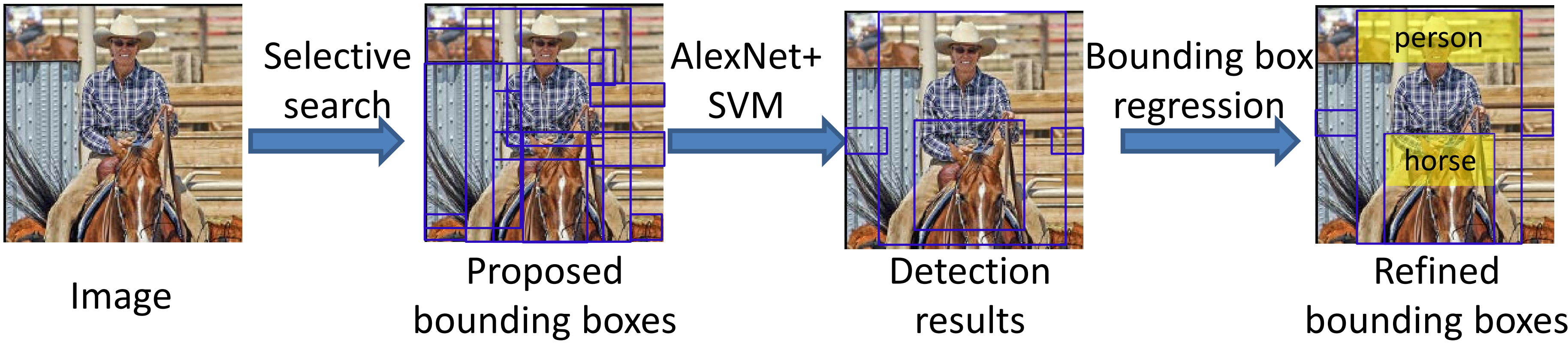
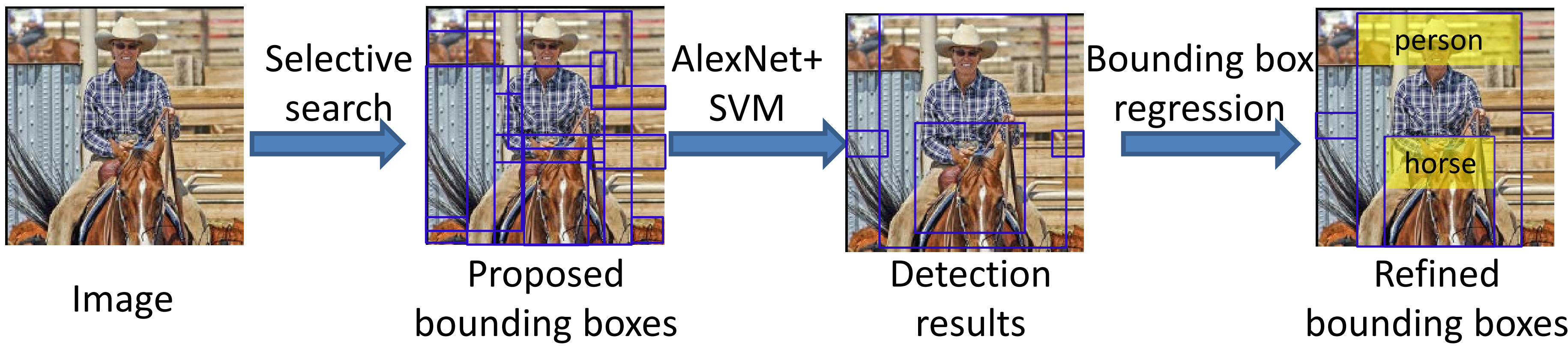


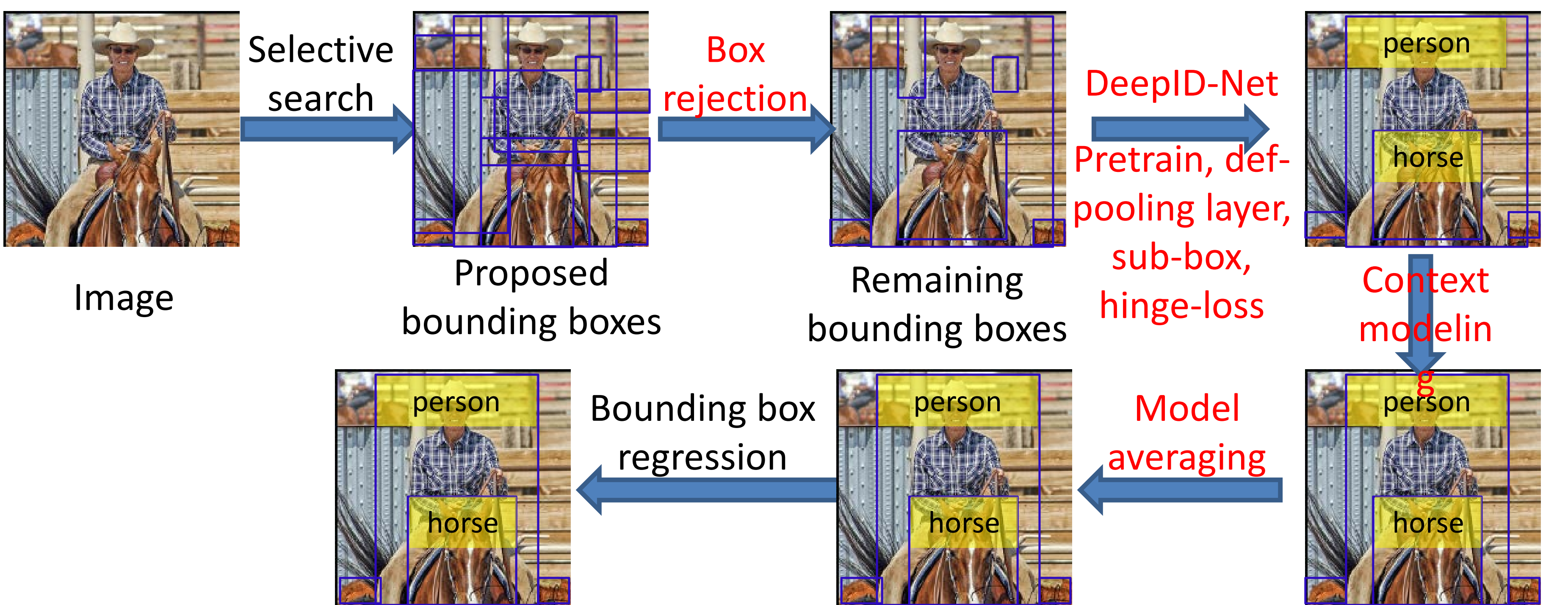
Image
Selective search
Proposed bounding boxes
AlexNet+ SVM
Detection results
Bounding box regression
person
horse
Refined bounding boxes

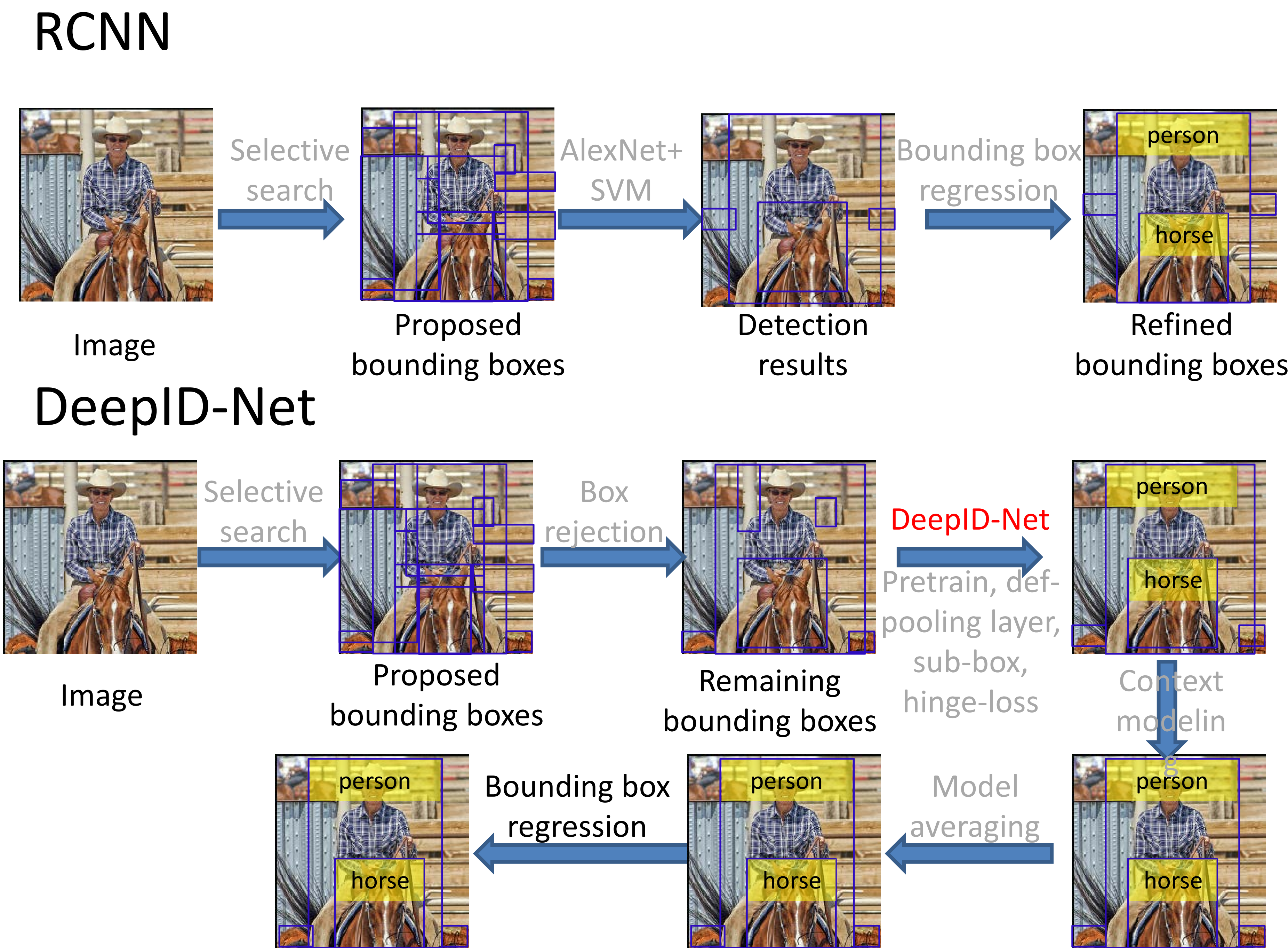
RCNN

Mean ap 31.4 → to 40.67 (new result on)

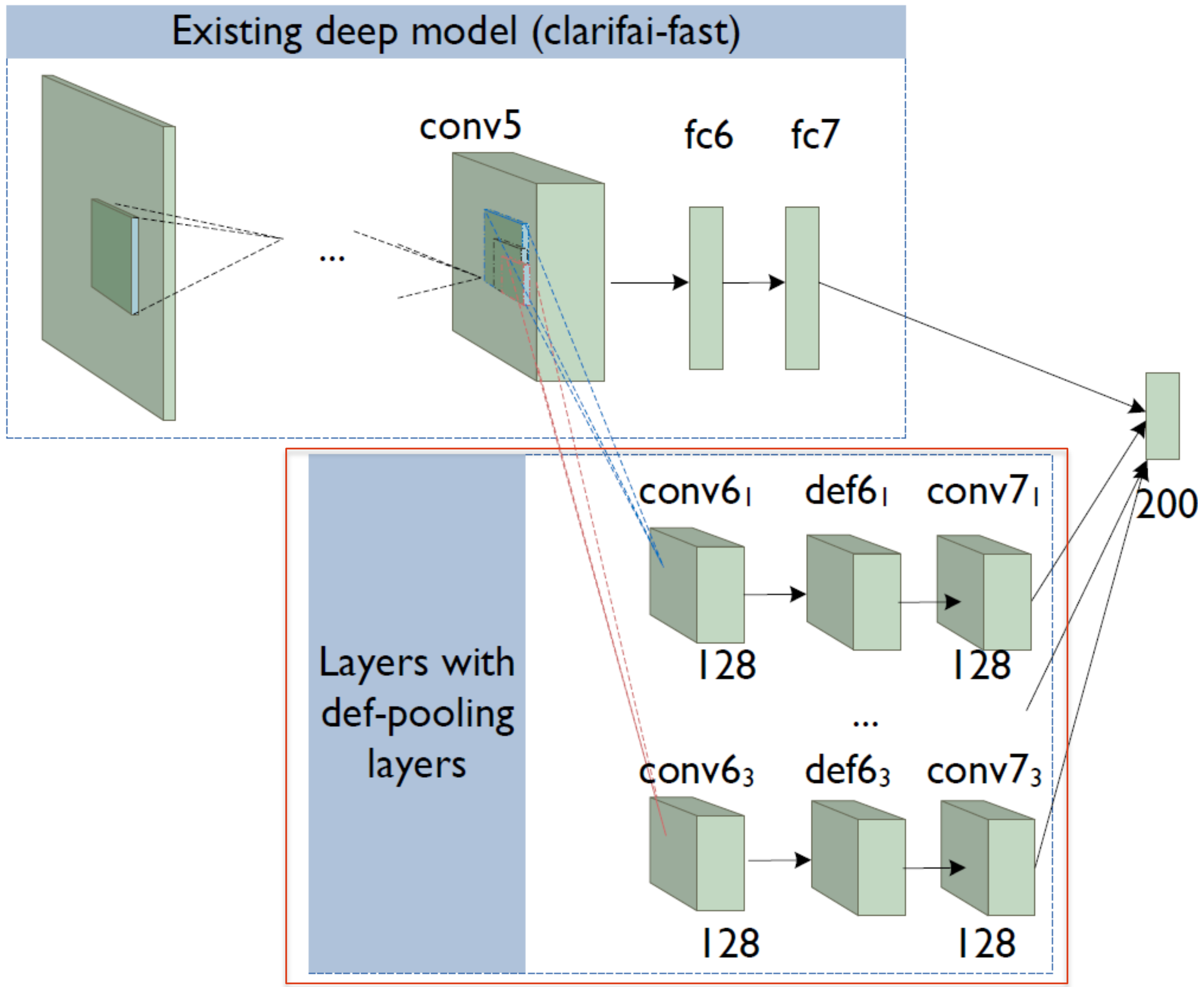


DeepID-Net



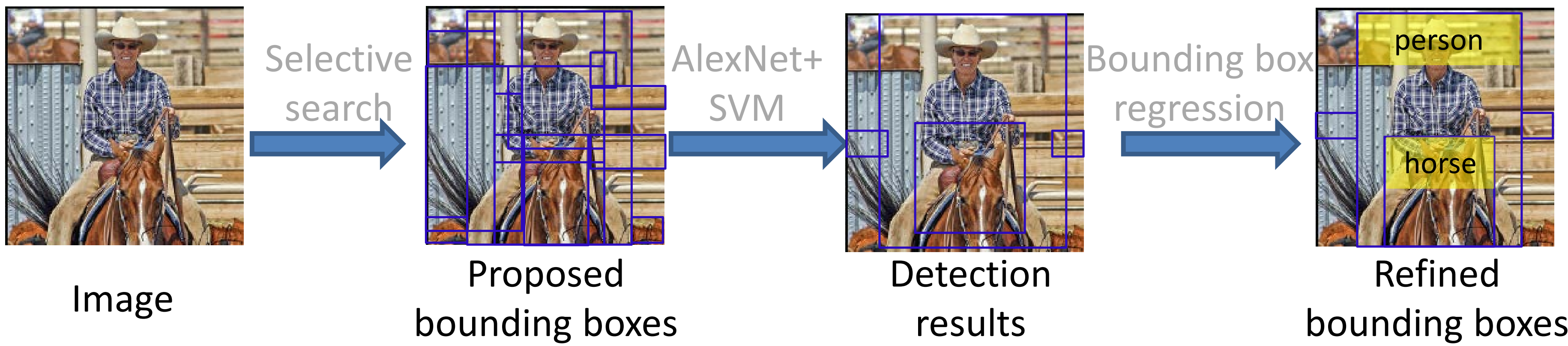

RCNN
Selective search
AlexNet+ SVM
Bounding box regression
person
horse
Image
Proposed bounding boxes
Detection results
Refined bounding boxes
DeepID-Net
Selective search
Box rejection
DeepID-Net
Pretrain, def-pooling layer, sub-box, hinge-loss
Image
Proposed bounding boxes
Remaining bounding boxes
Context modeling
Model averaging
Bounding box regression
person
horse

DeepID-Net

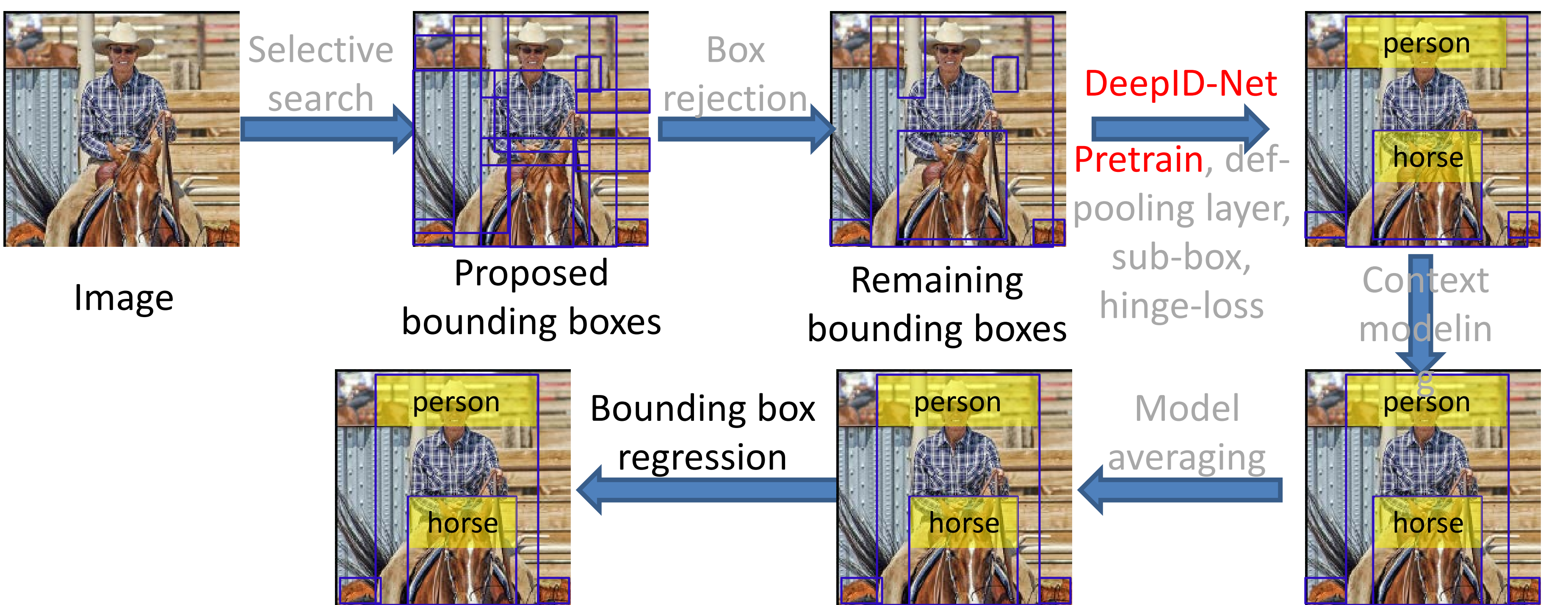


Existing deep model (clarifai-fast)
...
conv5
fc6
fc7
200
Layers with def-pooling layers
conv6₁
def6₁
conv7₁
128
128
...
conv6₃
def6₃
conv7₃
128
128

RCNN



DeepID-Net



Deep model training – pretrain

- RCNN (Cls+Det)
  - Pretrain on image-level annotation with 1000 classes
  - Finetune on object-level annotation with 200 classes
  - Gap: classification vs. detection, 1000 vs. 200

Image classification

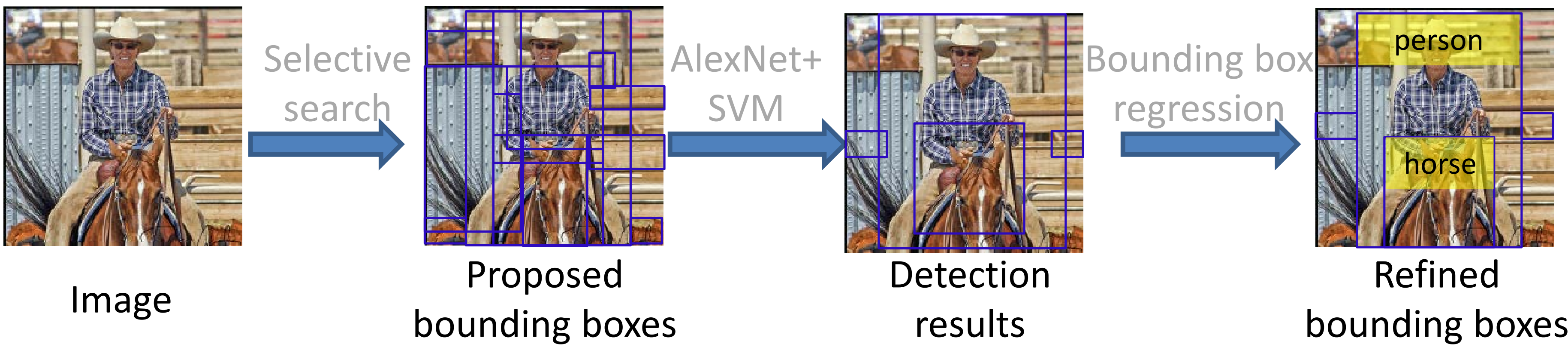
Object detection

Result and discussion

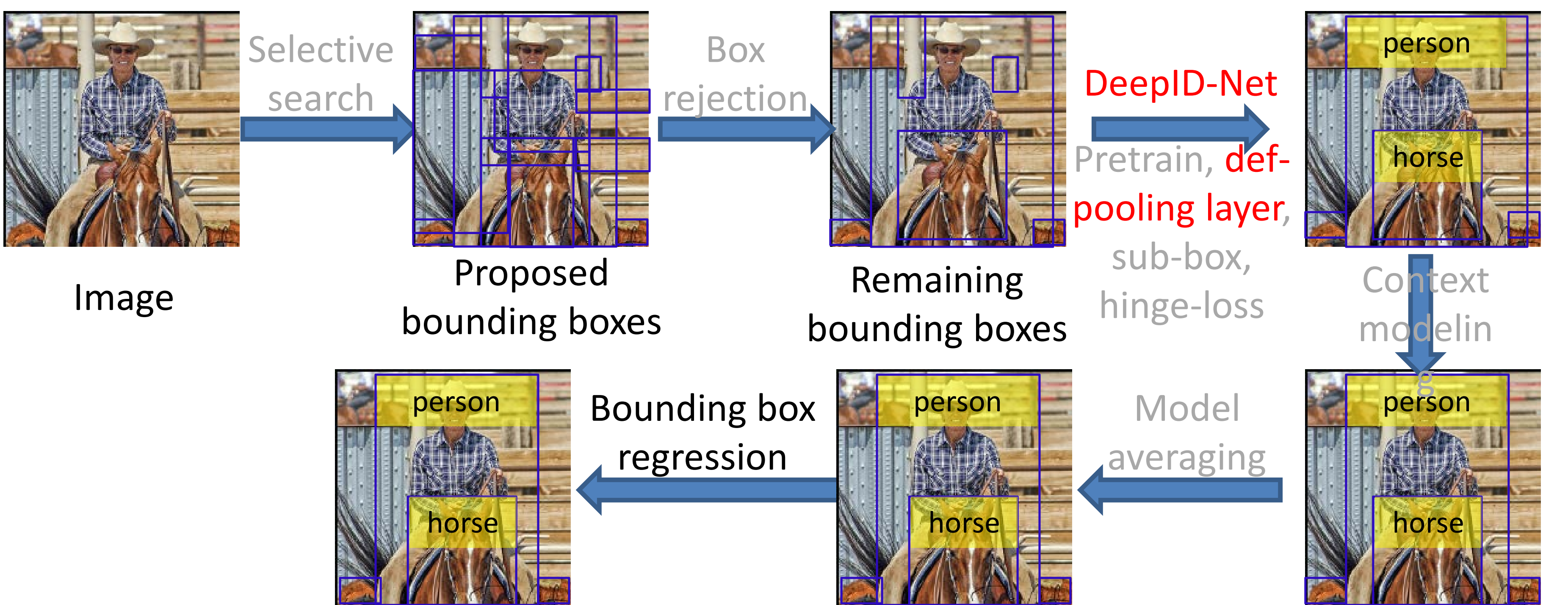
- Investigation
  - Better pretraining on 1000 classes
  - Object-level annotation is more suitable for pretraining
- Conclusions
  - The supervisory tasks should match at the pre-training and fine-turning stages
  - Although an application only involves detecting a small number of classes, it is better to pretraing with many classes outside the application

	Image annotation	Object annotation
200 classes (Det)	20.7	28.0
1000 classes (Cls-Loc)	31.8	36

RCNN



DeepID-Net



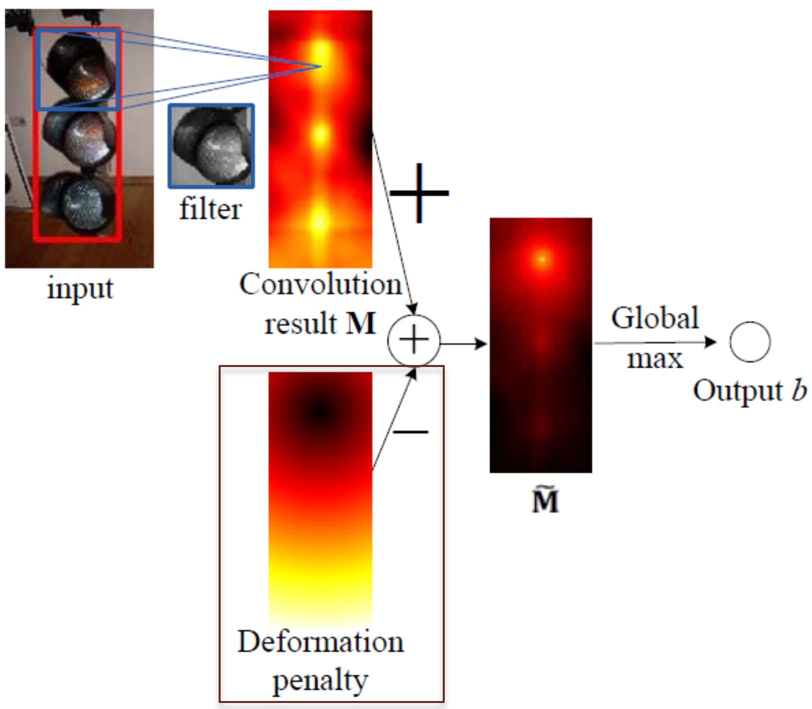
Deformation

– Learning deformation [a] is effective in computer vision society.
– Missing in deep model.
– We propose a new deformation constrained pooling layer.

[a] P. Felzenszwalb, R. B. Grishick, D.McAllister, and D. Ramanan. Object detection with discriminatively trained part based models. IEEE Trans. PAMI, 32:1627–1645, 2010.

Deformation Layer [b]

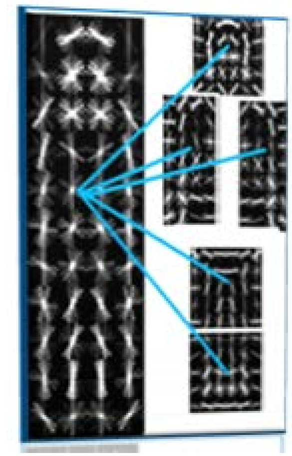
$$\mathbf{B}_p = \mathbf{M}_p + \boxed{\sum_{n=1}^{N} c_{n,p}\mathbf{D}_{n,p}} \qquad s_p = \max_{(x,y)} b_p^{(x,y)}$$



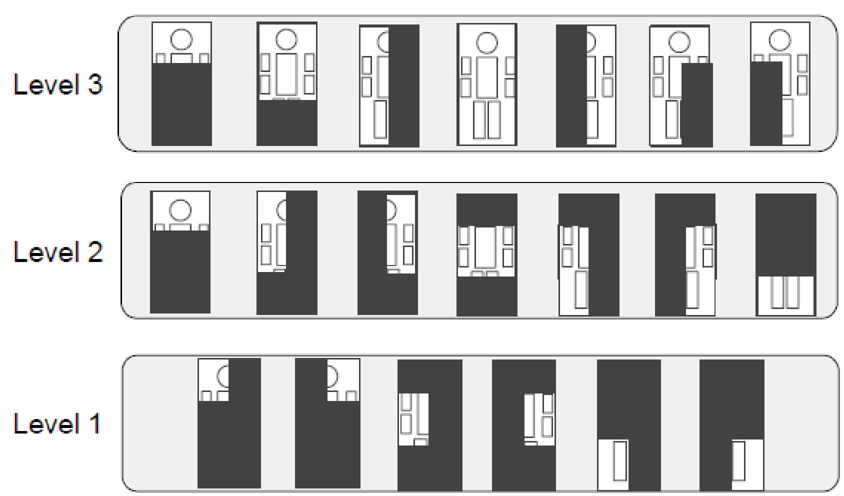
[b] Wanli Ouyang, Xiaogang Wang, "Joint Deep Learning for Pedestrian Detection ", ICCV 2013.

Modeling Part Detectors

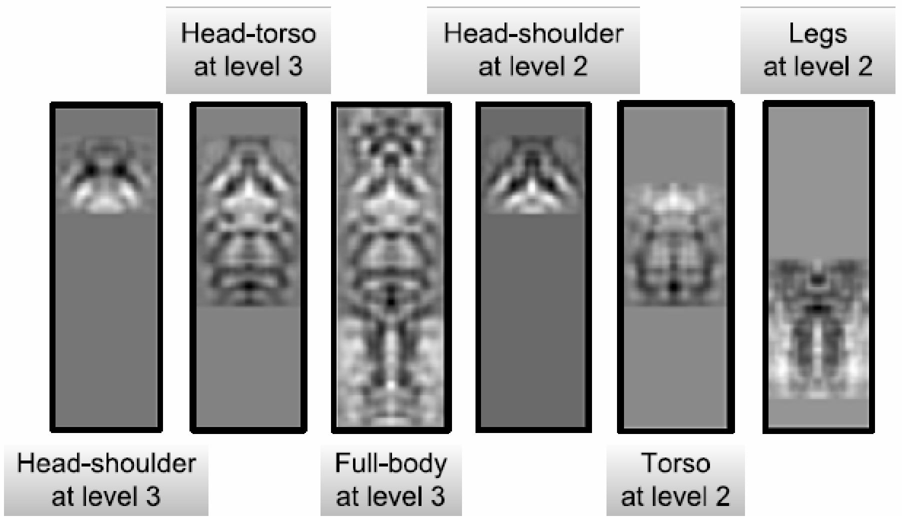
- Different parts have different sizes
- Design the filters with variable sizes



Part models learned from HOG



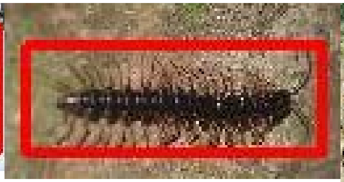
Part models



Learned filtered at the second convolutional layer

Deformation layer for repeated patterns

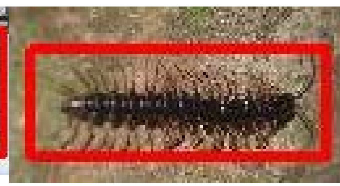
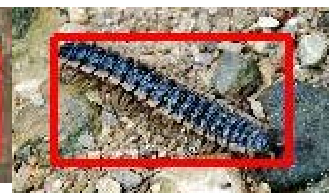
Pedestrian detection	General object detection
Assume no repeated pattern	Repeated patterns



Deformation layer for repeated patterns

Pedestrian detection	General object detection
Assume no repeated pattern	Repeated patterns
Only consider one object class	Patterns shared across different object classes



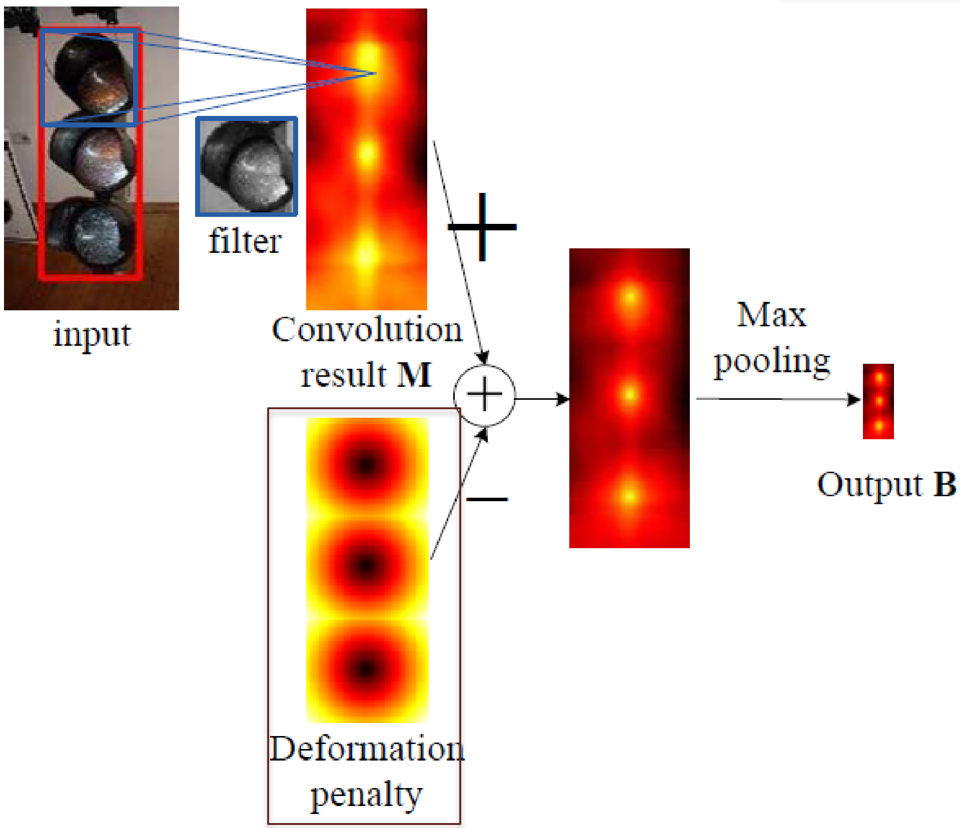


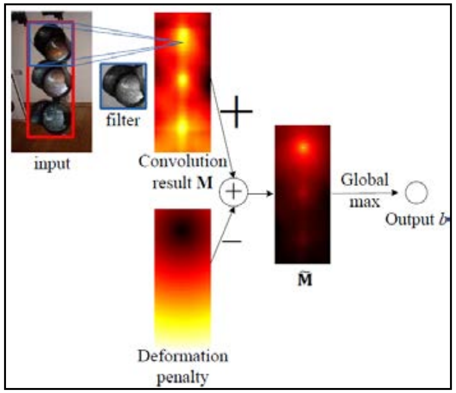


Deformation constrained pooling layer

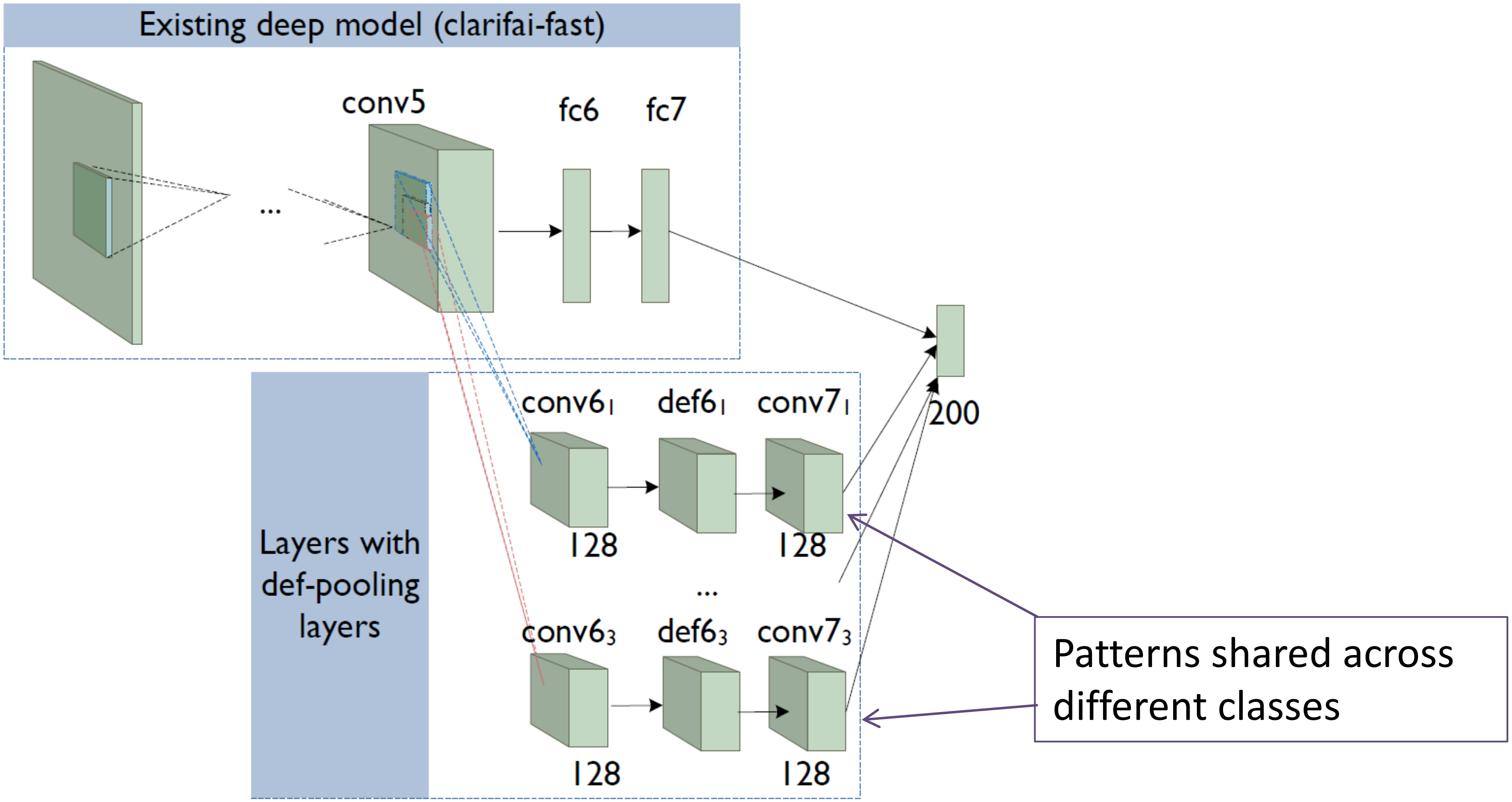
Can capture multiple patterns simultaneously

$$b^{(x,y)} = \max_{i,j \in \{-R,\cdots,R\}} \{m^{(k_x \cdot x+i, k_y \cdot y+j)} - \boxed{\sum_{n=1}^{N} c_n d_n^{i,j}}\},$$



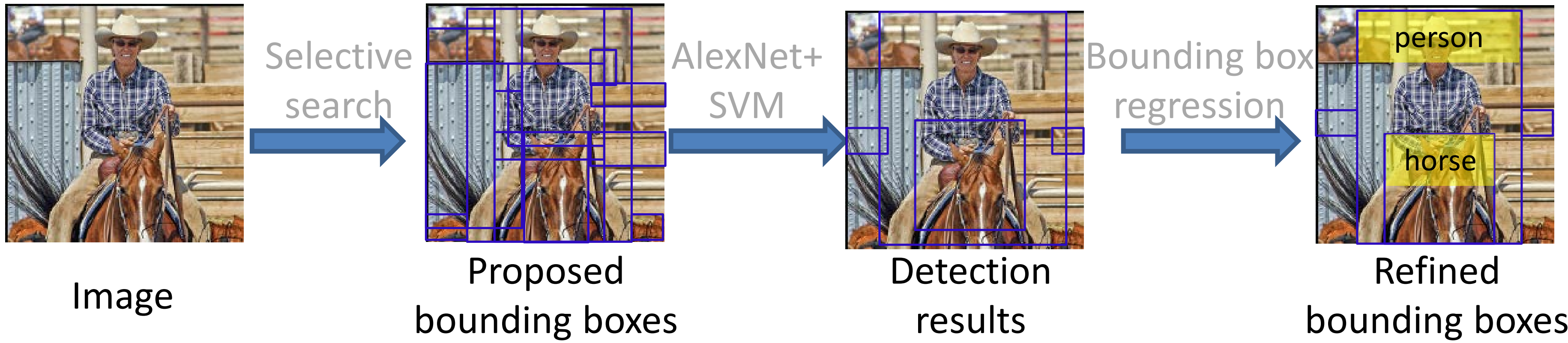


Our deep model with deformation layer

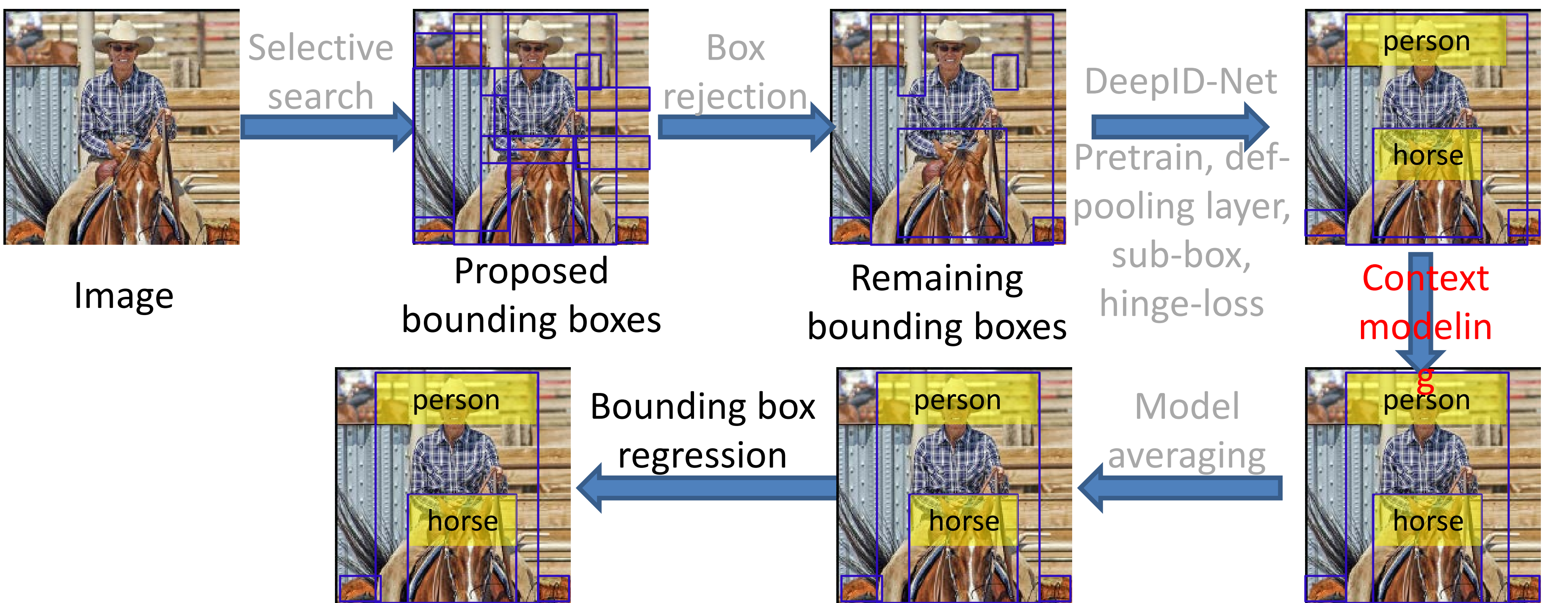


Training scheme	Cls+Det	Loc+Det	Loc+Det
Net structure	AlexNet	Clarifai	Clarifai+Def layer
Mean AP on val2	0.299	0.360	0.385

RCNN

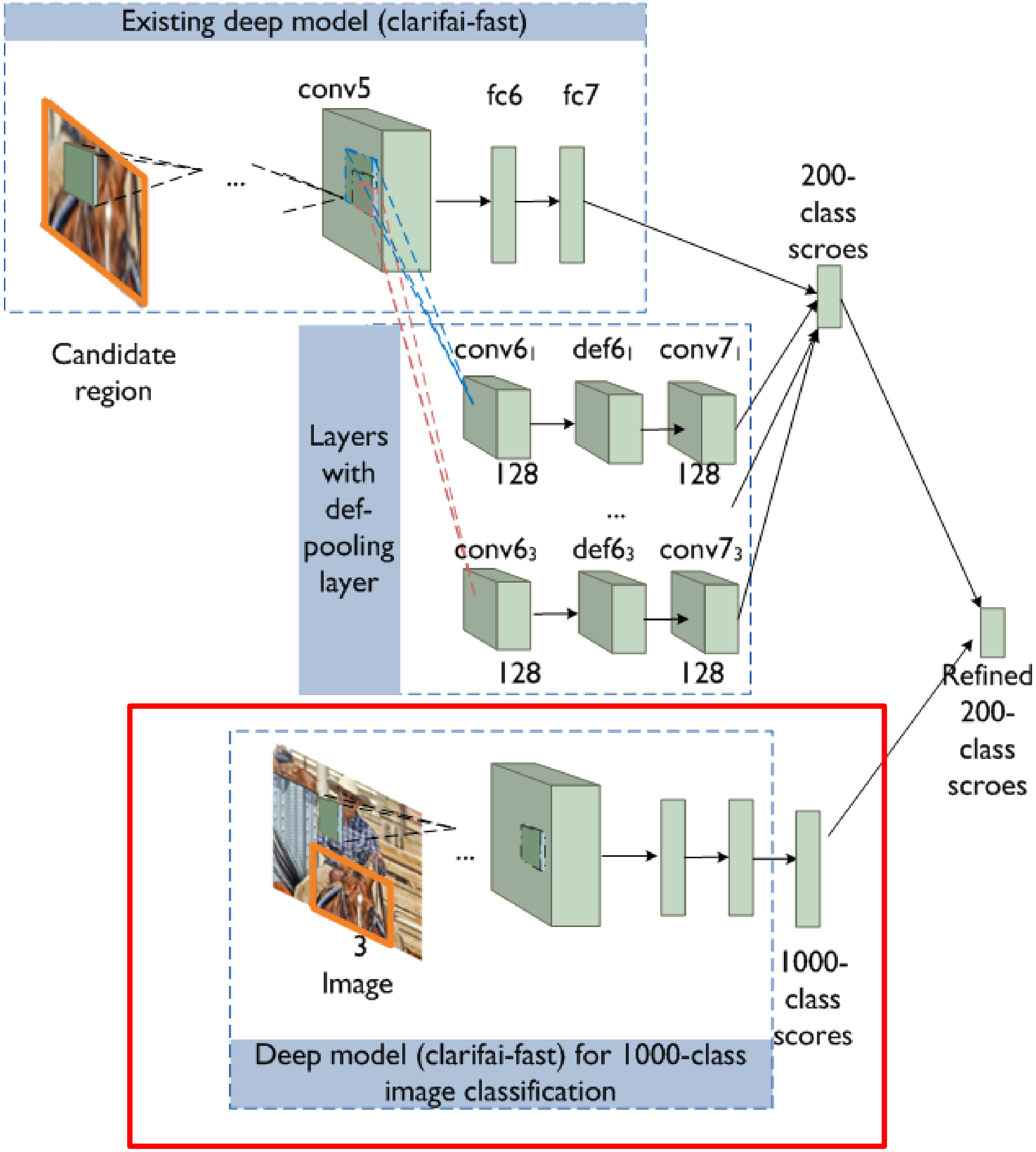


DeepID-Net



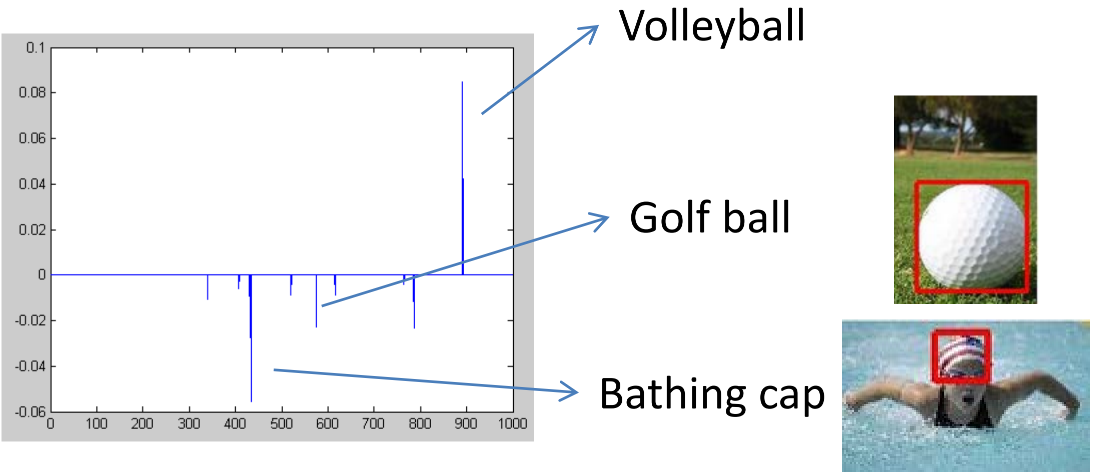
Context modeling

- Use the 1000 class Image classification score.
- ~1% mAP improvement.

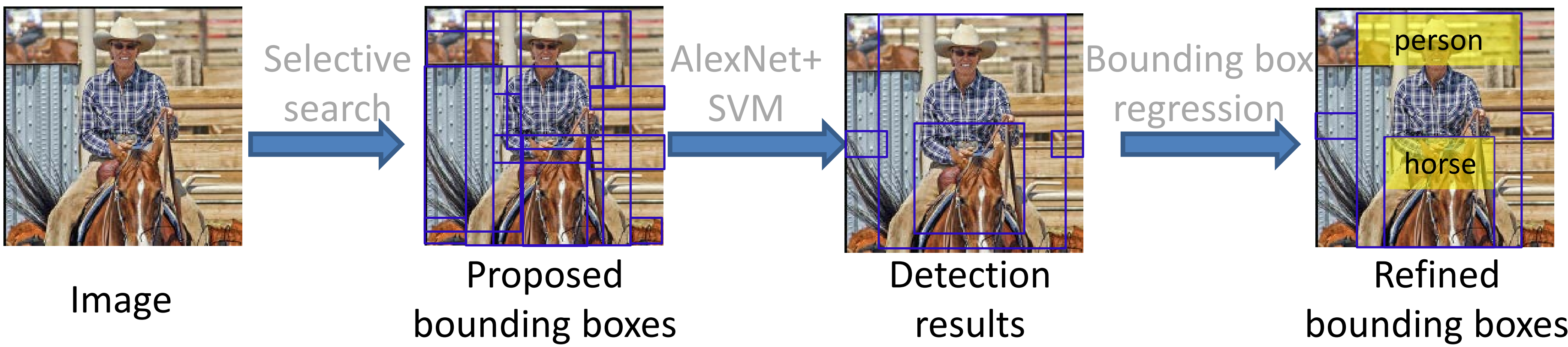


Context modeling

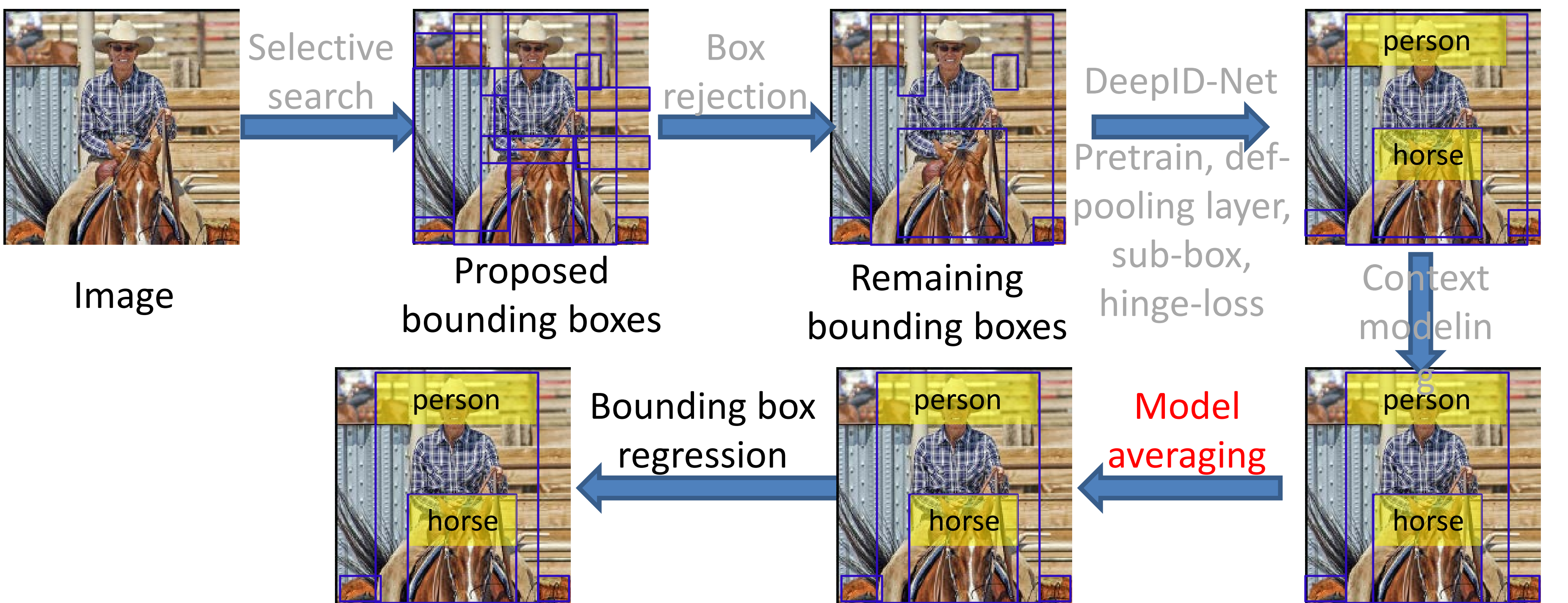
- Use the 1000-class Image classification score.
  - ~1% mAP improvement.
  - Volleyball: improve ap by 8.4% on val2.



RCNN



DeepID-Net

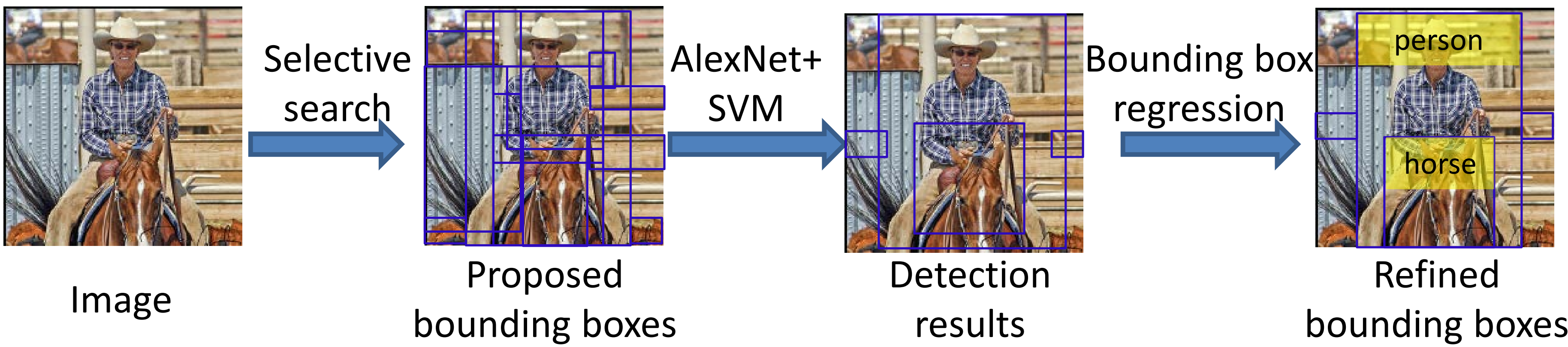


Model averaging

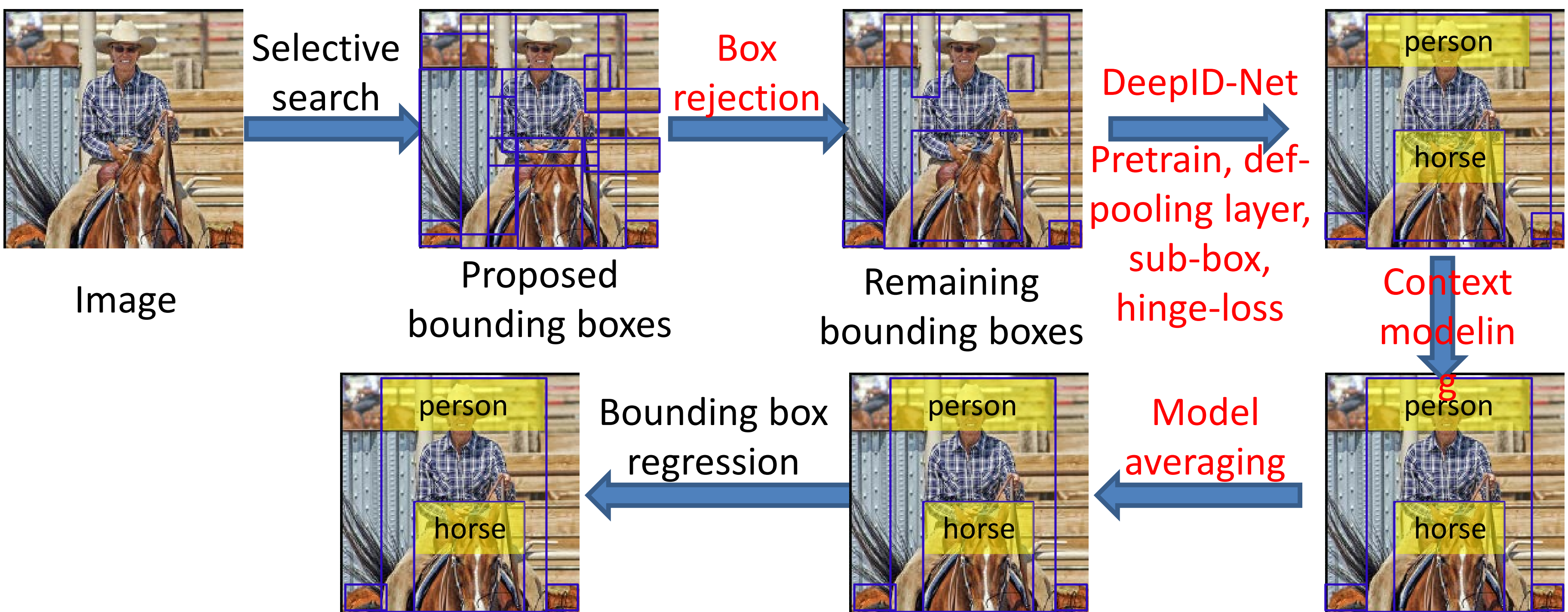
- Not only change parameters
  - Net structure: AlexNet(A), Clarifai (C), Deep-ID Net (D), DeepID Net2 (D2)
  - Pretrain: Classification (C), Localization (L)
  - Region rejection or not
  - Loss of net, softmax (S), Hinge loss (H)
  - Choose different sets of models for different object class

Model	1	2	3	4	5	6	7	8	9	10
Net structure	A	A	C	C	D	D	D2	D	D	D
Pretrain	C	C+L	C	C+L	C+L	C+L	L	L	L	L
Reject region?	Y	N	Y	Y	Y	Y	Y	Y	Y	Y
Loss of net	S	S	S	H	H	H	H	H	H	H
Mean ap	0.31	0.312	0.321	0.336	0.353	0.36	0.37	0.37	0.371	0.374

RCNN



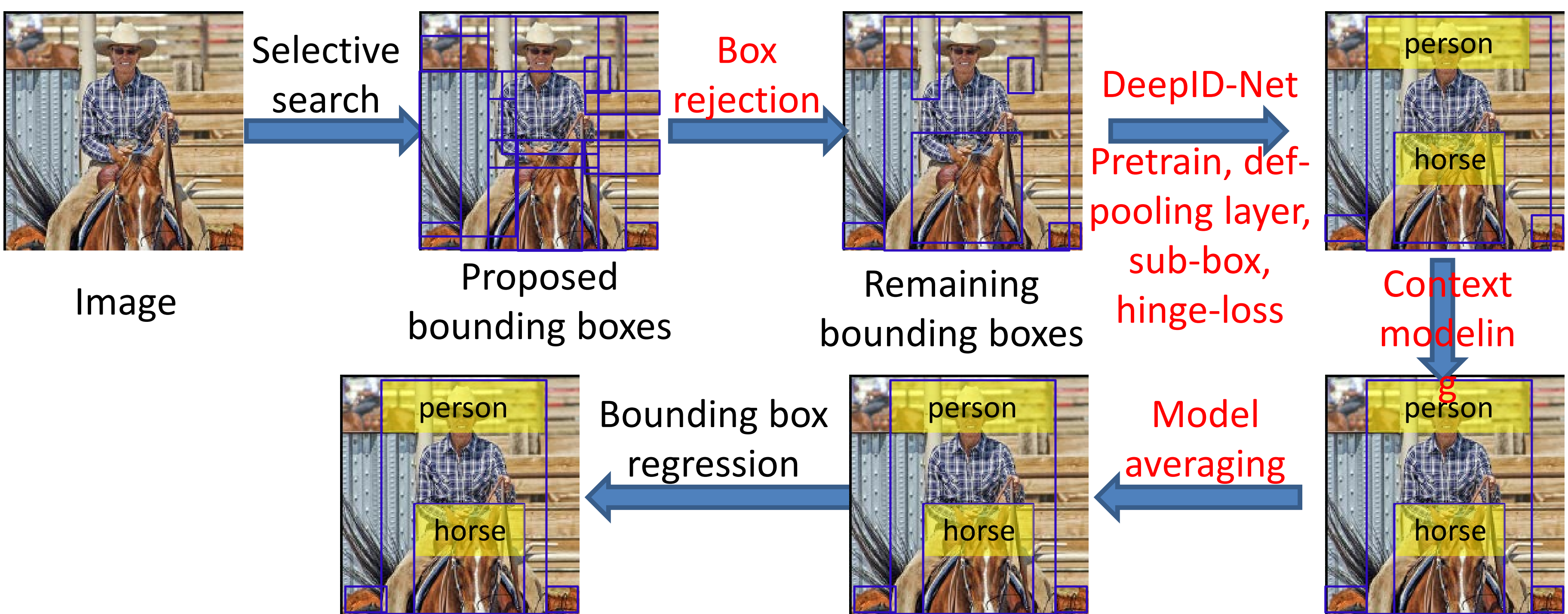
DeepID-Net



Component analysis

Detection Pipeline	RCNN	Box rejection	Clarifai	Loc+ Det	+Def layer	+cont ext	+bbox regr.	Model avg.
mAP on val2	29.9	30.9	31.8	36.0	38.5	39.2	40.1	42.4
mAP on test					38.0	38.6	39.4	41.7

DeepID-Net



Summary

- Bounding rejection. Save feature extraction by about 10 times, slightly improve mAP (~1%).
- Pre-training with object-level annotation, more classes. 4.2% mAP
- Def-pooling layer. 2.5% mAP improvement
- Contextual modeling. 1% mAP improvement
- Model averaging. 2.3% mAP improvement. Different model designs and training schemes lead to high diversity

Reference

- R. Girshick, J. Donahue, T. Darrell, J. Malik, “Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation,” CVPR, 2014.
- Sermanet, Pierre, et al. "Overfeat: Integrated recognition, localization and detection using convolutional networks." *arXiv preprint arXiv:1312.6229* (2013).
- W. Ouyang and X. Wang, "Joint Deep Learning for Pedestrian Detection," ICCV 2013
- X. Zeng, W. Ouyang and X. Wang, "Multi-Stage Contextual Deep Learning for Pedestrian Detection," ICCV 2013
- P. Luo, Y. Tian, X. Wang, and X. Tang, "Switchable Deep Network for Pedestrian Detection", CVPR 2014
- W. Ouyang, X. Zeng and X. Wang, "Modeling Mutual Visibility Relationship with a Deep Model in Pedestrian Detection," CVPR 2013
- W. Ouyang, and X. Wang, "A Discriminative Deep Model for Pedestrian Detection with Occlusion Handling," CVPR 2012
- Y. Sun, X. Wang and X. Tang, "Deep Convolutional Network Cascade for Facial Point Detection," CVPR 2013
- W. Ouyang, X. Chu, and X. Wang, "Multi-source Deep Learning for Human Pose Estimation", CVPR 2014
- Y. Yang and D. Ramanan, “Articulated Pose Estimation with Flexible Mixtures-of-Parts,” CVPR 2011.

Reference

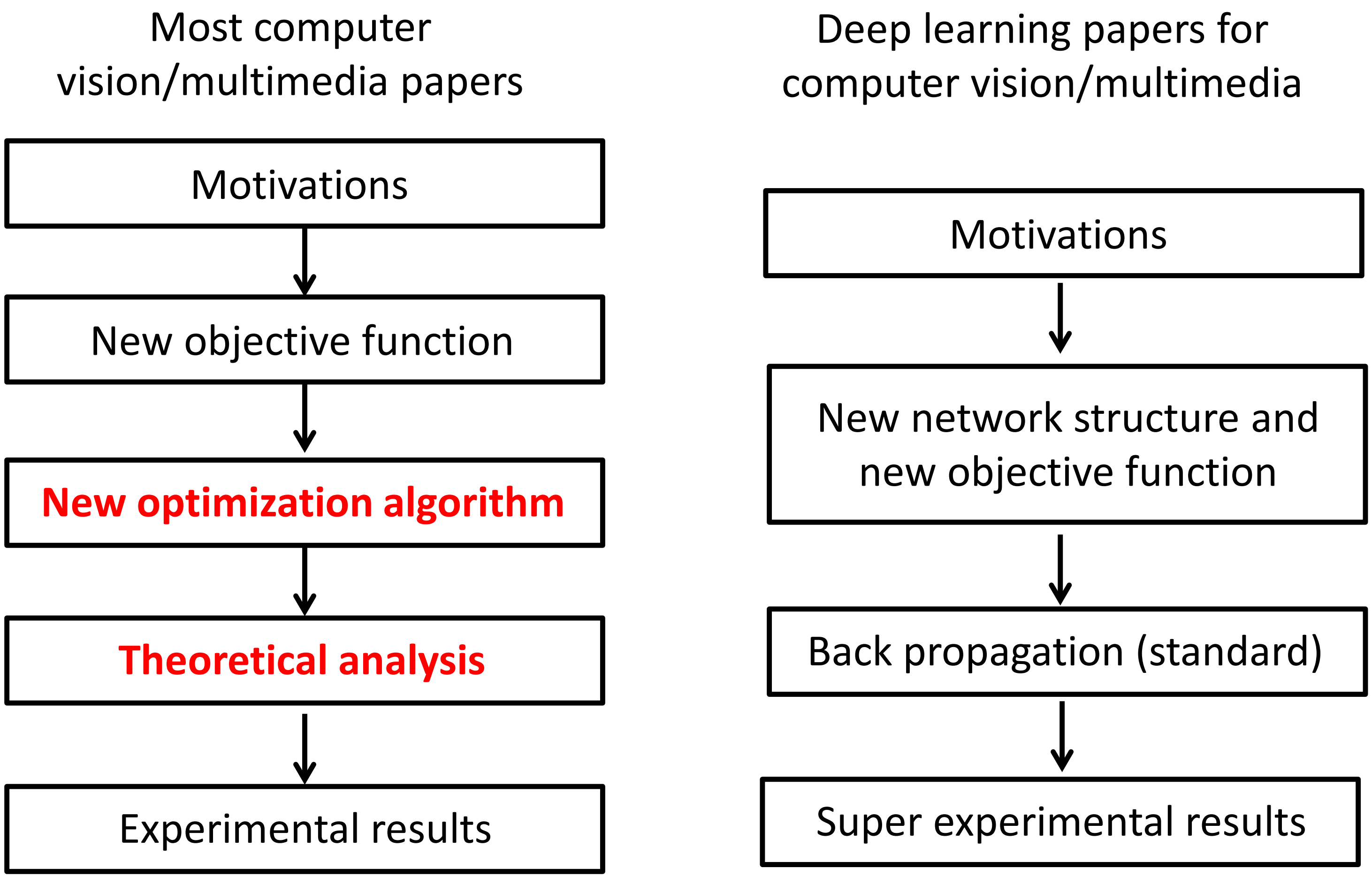
- X. Wang, M. Yang, S. Zhu, and Y. Lin, “Regionlets for Generic Object Detection,” ICCV 2013.
- S. Fidler, R. Mottaghi, A. Yuille, and R. Urtasun, “Bottom-up Segmentation for Top-Down Detection,” CVPR 2013.
- A. Krizhevsky, L. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” Proc. NIPS, 2012.
- J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, A. W. M. Smeulders, “Selective Search for Object Recognition,” IJCV 2013.
- W. Ouyang and X. Wang, “DeepID-Net: deformable deep convolutional neural networks for object detection,” CVPR, 2015.

Outline

- Introduction to deep learning
- Deep learning for object recognition
- Deep learning for object segmentation
- Deep learning for object detection
- **Open questions and future works**

“Concerns” on deep learning

- C1: Weak on theoretical support (convergence, bound, local minimum, why it works)
  - It’s true. That’s why deep learning papers were not accepted by the computer vision/image processing community for a long time. Any theoretical studies in the future are important.



That’s probably one of the reasons that computer vision and image processing people think deep learning papers are lack of novelty and theoretical contribution ☹

“Concerns” on deep learning

- C2: It is hard for computer vision/image processing people to have innovative contributions to deep learning. Our job becomes preparing the data + using deep learning as a black box. That’s the end of our research life.
  - That’s not true. Computer vision and image processing researchers have developed many systems with deep architectures. But we just didn’t know how to jointly learn all the components. Our research experience and insights can help to design new deep models and pre-training strategies.
  - Many machine learning models and algorithms were motivated by computer vision and image processing applications. However, computer vision and multimedia did not have close interaction with neural networks in the past 15 years. We expect fast development of deep learning driven by applications.

“Concerns” on deep learning

- C3: Since the goal of neural networks is to solve the general learning problem, why do we need domain knowledge?
  – The most successful deep model on image and video related applications is convolutional neural network, which has used domain knowledge (filtering, pooling)
  – Domain knowledge is important especially when the training data is not large enough

“Concerns” on deep learning

- C4: Good results achieved by deep learning come from manually tuning network structures and learning rates, and trying different initializations
  - That’s not true. One round evaluation may take several weeks. There is no time to test all the settings.
  - Designing and training deep models does require a lot of empirical experience and insights. There are also a lot of tricks and guidance provided by deep learning researchers. Most of them make sense intuitively but without strict proof.

"Concerns" on deep learning

- C5: Deep learning is more suitable for industry rather than research groups in universities
  - Industry has big data and computation resources
  - Research groups from universities can contribute on model design, training algorithms and new applications

“Concerns” on deep learning

- C6: Deep learning has different behaviors when the scale of training data is different
  - Pre-training is useful when the training data small, but does not make big difference when the training data is large enough
  - So far, the performance of deep learning keep increasing with the size of training data. We don’t see its limit yet.
  - Shall we spend more effort on data annotation or model design?

Future works

- Explore deep learning in new applications
  - Worthy to try if the applications require features or learning, and have enough training data
  - We once had many doubts on deep. (Does it work for vision? Does it work for segmentation? Does it work for low-level vision?) But deep learning has given us a lot of surprises.
  - Applications will inspire many new deep models
- Incorporate domain knowledge into deep learning
- Integrate existing machine learning models with deep learning

Future works

- Deep learning to extract dynamic features for video analysis
- Deep models for structured data
- Theoretical studies on deep learning
- Quantitative analysis on how to design network structures and how to choose nonlinear operations of different layers in order to achieve feature invariance
- New optimization and training algorithms
- Parallel computing systems and algorithm to train very large and deep networks with larger training data

Home Join Us People Publications **Projects** Datasets Lab Activities Intranet Contact Us

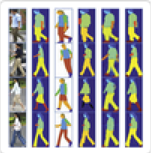
MMLAB

Multimedia Laboratory

Projects / Deep Learning

Introduction Publications **Codes** Slides Deep Learning Bibliography Useful Links

Description **Download**



A demo code that allows you to input a pedestrian image and then compute the label map. — Zip

Reference:

1. P. Luo, X. Wang, and X. Tang, "Pedestrian Parsing via Deep Decompositional Neural Network," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)* 2013 [PDF] [Project Page]

A demo code that shows you how the frontal-view face image of a query face image is reconstructed. — Zip

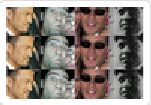
Reference:

1. Z. Zhu, P. Luo, X. Wang, and X. Tang, "Deep Learning Identity Preserving Face Space," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)* 2013 [PDF] [Project Page]

Matlab training and testing source code for pedestrian detection using the proposed approach. Models trained on INRIA and Caltech are provided. — Webpage

Reference:

1. Wanli Ouyang, Xiaogang Wang, "Joint Deep Learning for Pedestrian Detection", in *Proceedings of IEEE International Conference on Computer Vision (ICCV)* 2013 [PDF] [Project Page]
2. Wanli Ouyang, Xiaogang Wang, "A Discriminative Deep Model for Pedestrian Detection with Occlusion Handling", in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 2012 [PDF] [Project Page]



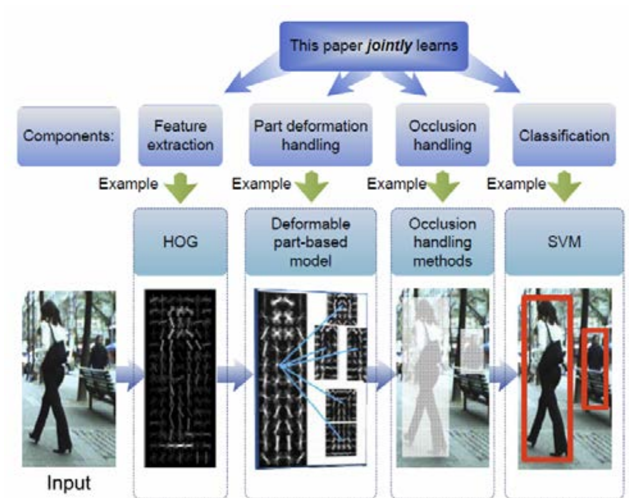
Executable files for the face detector and facial point detector. — Webpage

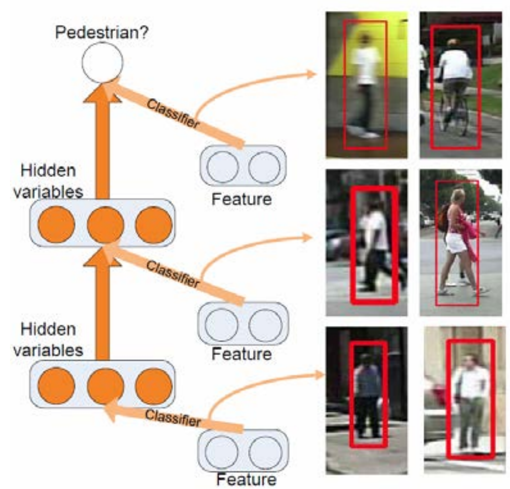
Reference:

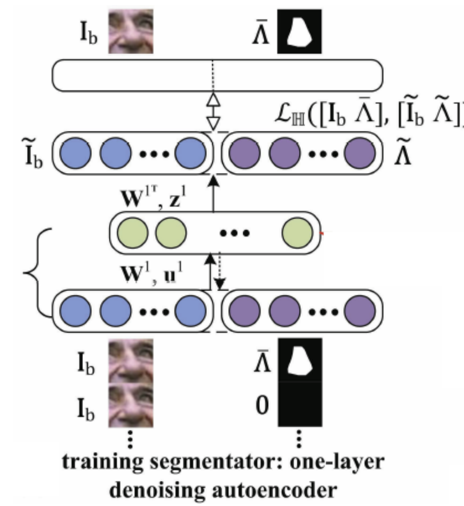
1. Y. Sun, X. Wang and X. Tang, "Deep Convolutional Network Cascade for Facial Point Detection," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3476-3483, 2013 [PDF] [Project Page]

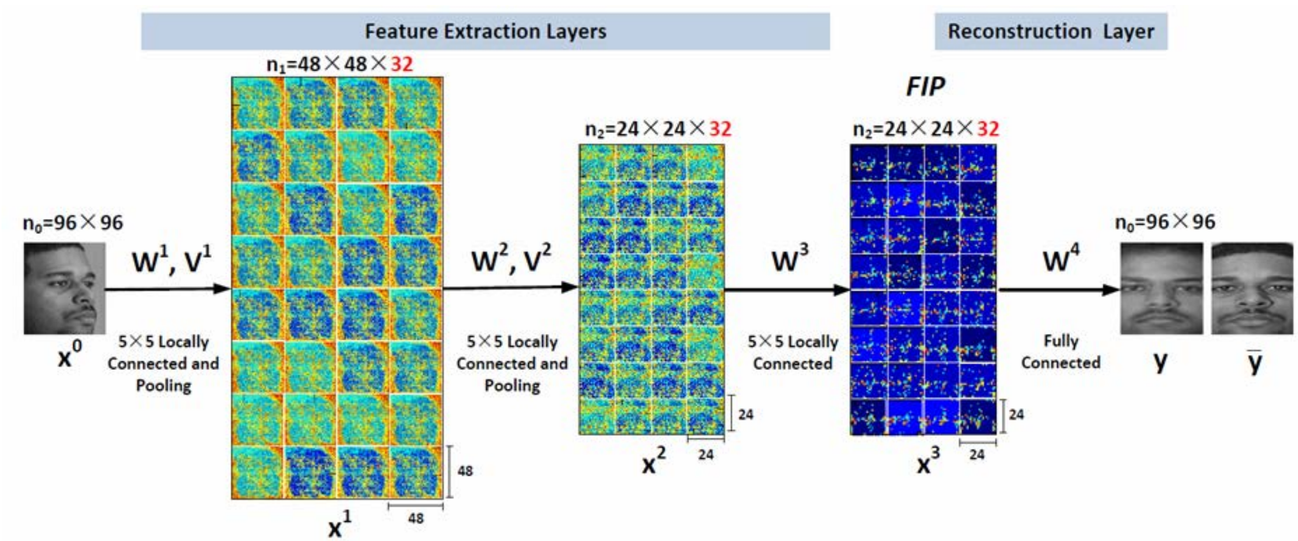
http://mmlab.ie.cuhk.edu.hk/project_deep_learning.html

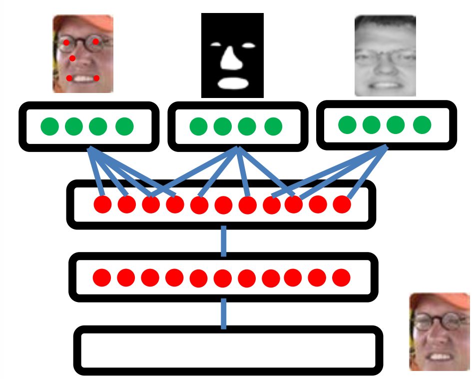
Thank you!











http://mmlab.ie.cuhk.edu.hk/ http://www.ee.cuhk.edu.hk/~xgwang/