MSPR8 Convolutional Neural Networks

Dr. Hendrik Purwins,
Aalborg University Copenhagen

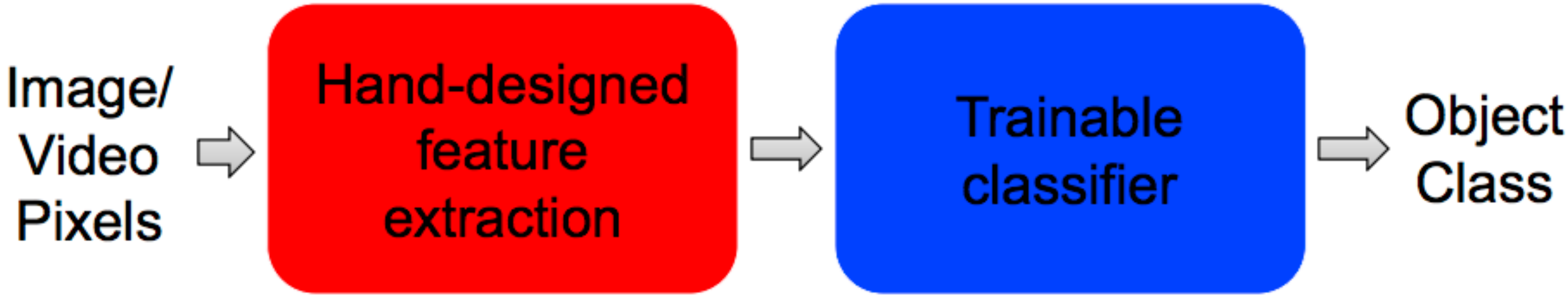
Automatic Reading of Handwritten Postcodes

- MNIST collection of 70000 hand-written digits
- Download t10k-images-idx3-ubyte and t10k-labels-idx1-ubyte and cnn0.83.zip from Moodle
- Unpack CNN (Library for Convolutional Neural Networks) and add its path and subpath in Matlab.

Reading and Displaying Images from the Handwritten Digits Database

- Display the first two images of the MNST test data base and get their annotated labels
- Load 2 images into your workspace:
- I=readMNIST_image([MNISTpath testFname],2);
- Reformat the images so their grey values are within 0 and 1:
- imdisp = abs(double(I{1})/255-1)';
- Reshape the images into a 28x28 format and display them:
- imagesc(reshape(imdisp,28,28));
- Fix the greyscale colormap:
- colormap([0:1/255:1;0:1/255:1;0:1/255:1]');

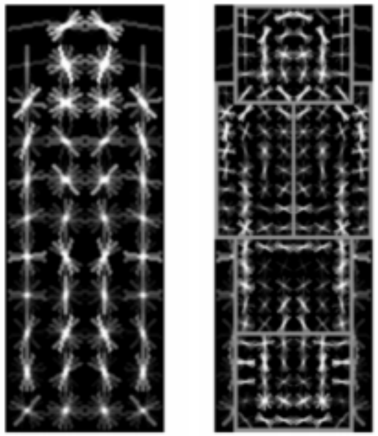
Traditional Recognition Approach



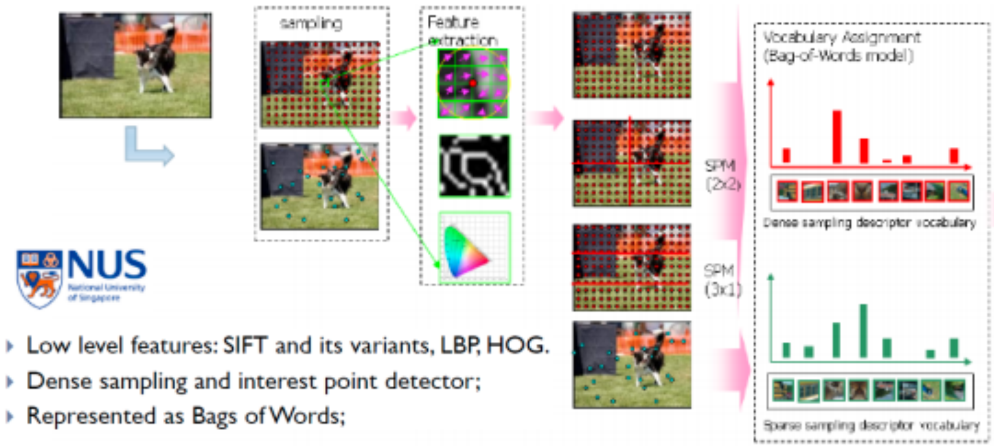
- Features are not learned

Traditional Recognition Approach

- Features are key to recent progress in recognition
- Multitude of hand-designed features currently in use
  - SIFT, HOG, ...
- Where next? Better classifiers? Or keep building more features?



Felzenszwalb, Girshick, McAllester and Ramanan, PAMI 2007



Yan & Huang
(Winner of PASCAL 2010 classification competition)

Convolution (1D)

- Correlation over a Window M:

$$corr(f,g)(n) = \sum_{m=0}^{M-1} f(n-m) \cdot g(n-m)$$

- Convolution (correlation with flipped g):

$$(f * g)(n) = \sum_{m=0}^{M-1} f(n-m) \cdot g(m)$$

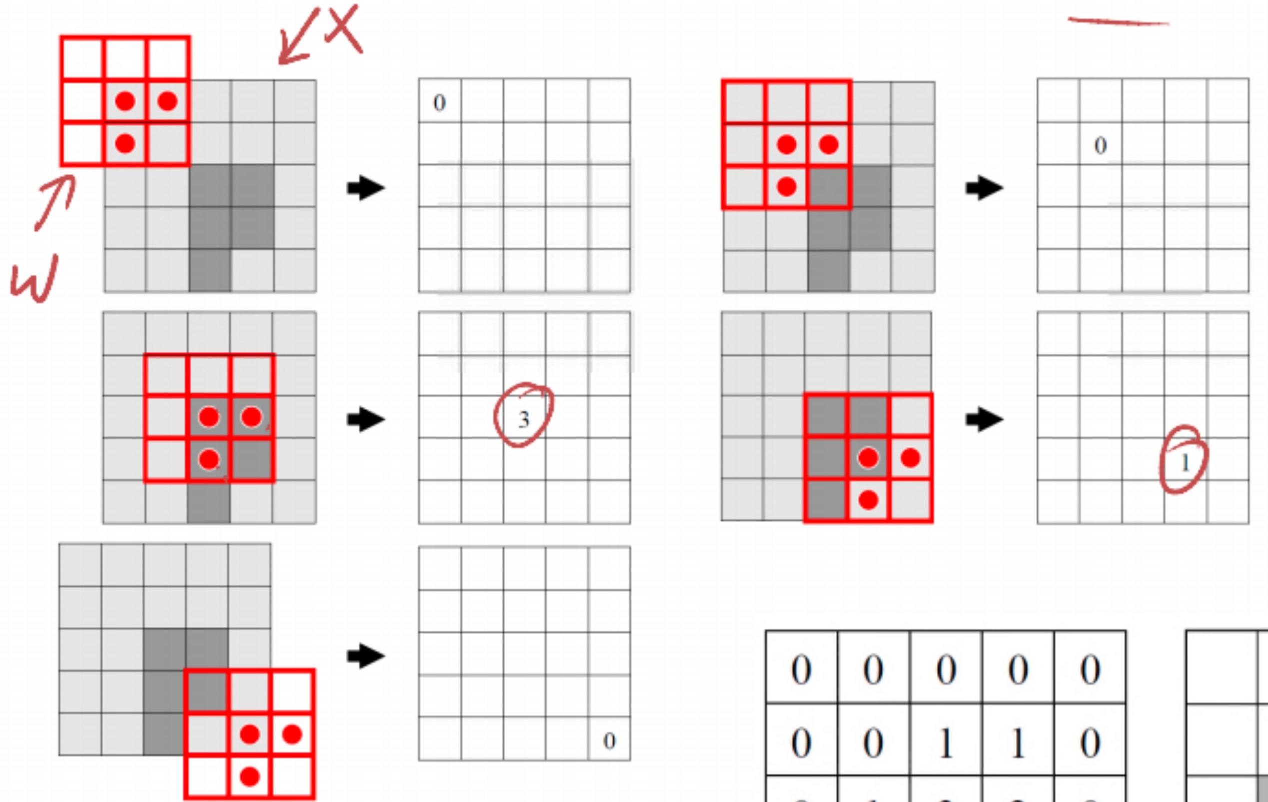
Matlab: conv(f,g)

Convolution Exercise

- Load the file Elvis.wav
- Use the Matlab functions conv to convolve the sound file with 4 periods of sine wave of 108 Hz frequence, sampled at the same frequence as Elvis.
- Listen and plot the result, plot the orginal sound in the same plot with another color.
- Interprete the results!
- (for SMC:)Construct an FIR filter using the sine wave as the filter coefficients. Compare the results.

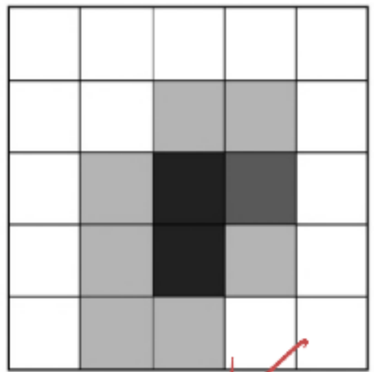
Image convolution

For the image, take dark pixel value = 1, light pixel value = 0.



feature map

0	0	0	0	0
0	0	1	1	0
0	1	3	2	0
0	1	3	1	0
0	1	1	0	0



Leow Wee Kheng

https://www.cs.ox.ac.uk/people/nando.defreitas/machinelearning/lecture9.pdf

Convolution (2D)

- Convolve matrix A with matrix B:

Matlab: `conv2(A,B)`

Class Assignment

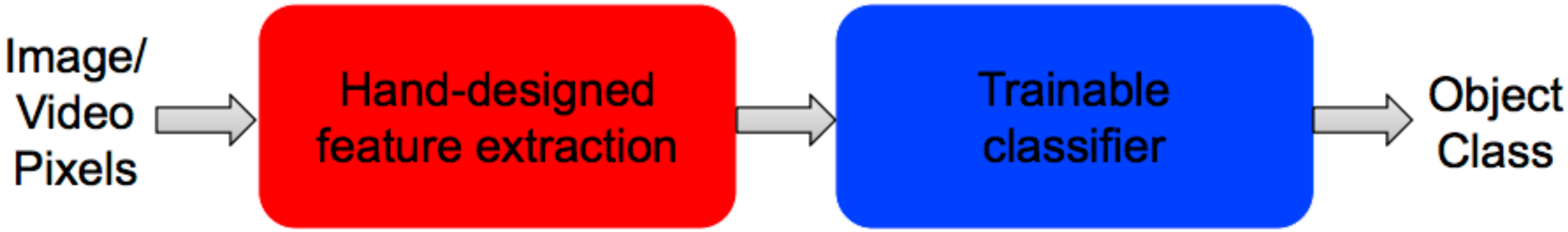
- Get Image 2 from the MNIST Test data base and plot it.
- Convolve it with matrix $B = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$
- Plot the convolved image.
- What can you observe?

What about learning the features?

- Learn a *feature hierarchy* all the way from pixels to classifier
- Each layer extracts features from the output of previous layer
- Train all layers jointly

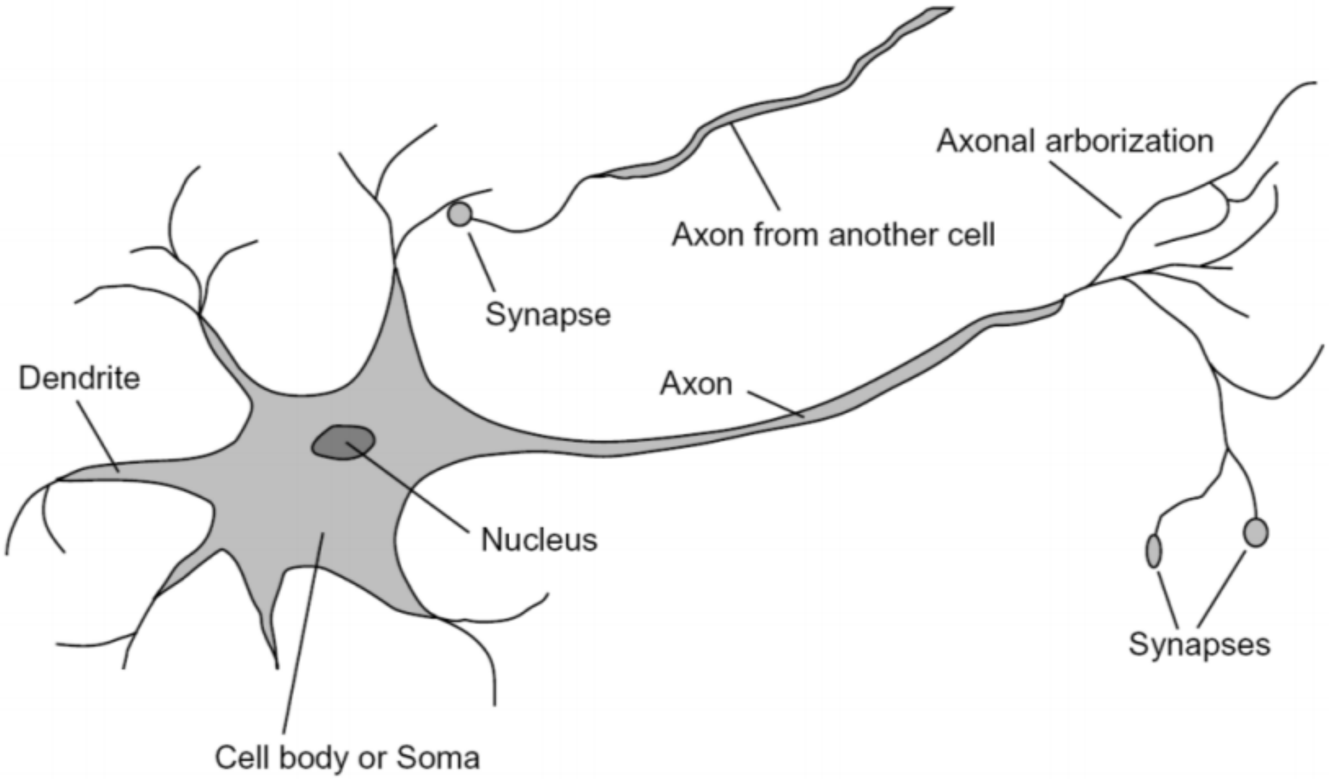
“Shallow” vs. “deep” architectures

Traditional recognition: “Shallow” architecture

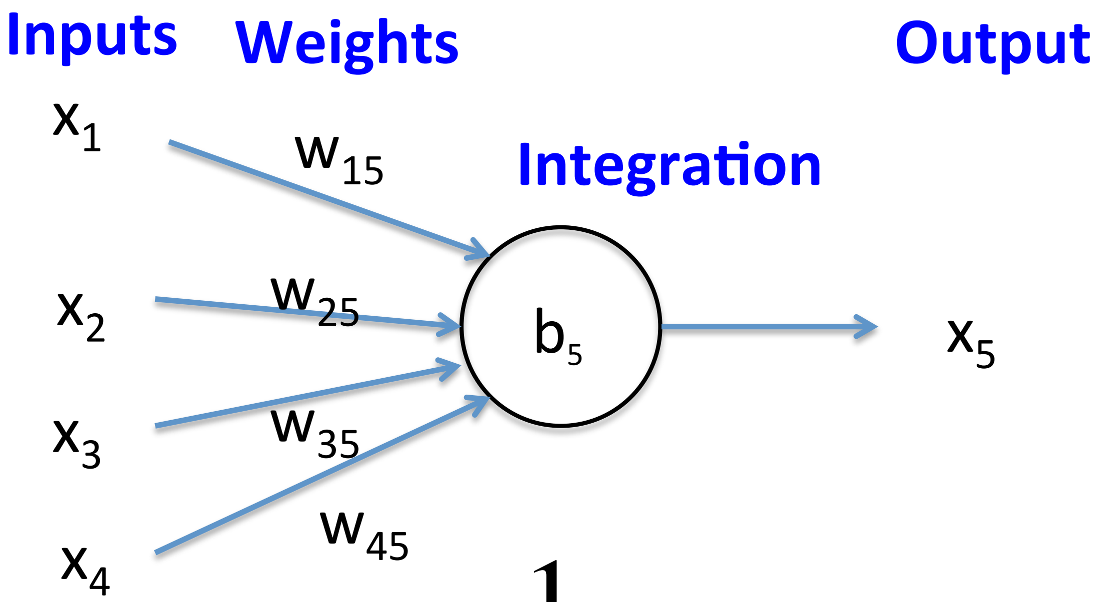


Deep learning: “Deep” architecture

Inspiration: Neuron cells



Feedforward Activation in Artificial Neural Networks: Perceptron (Rosenblatt 1957)

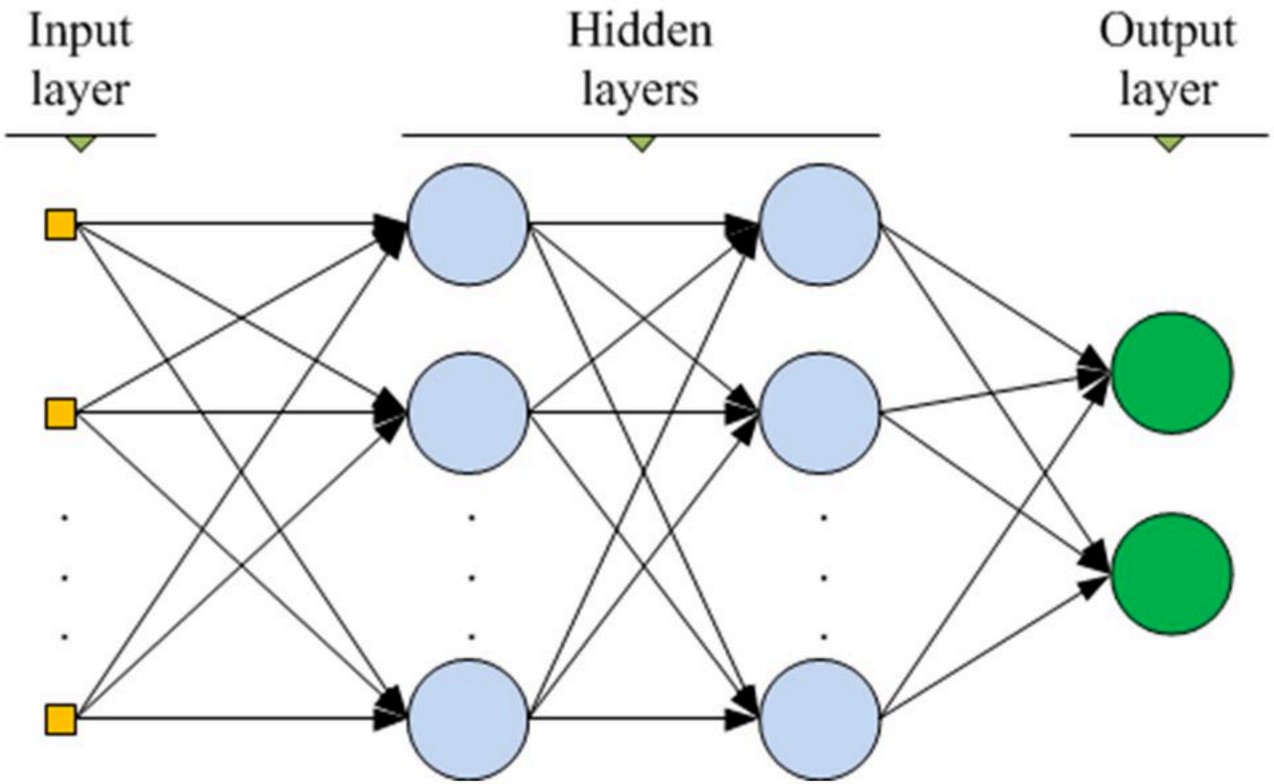


$$\frac{1}{1+e^{-(b_5+\sum_{i=1}^{4} w_{i5} x_i)}} = x_5$$

Feedforward Activation in Artificial Neural Networks: Perceptron (Rosenblatt 1957)

- "the embryo of an electronic computer that [the Navy] expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence.”
- Minsky/ Papert (1969): cannot even learn XOR function!!
- BUT: Multilayer perceptron!

Feedforward Activation in Artificial Neural Networks: Multilayer Perceptron

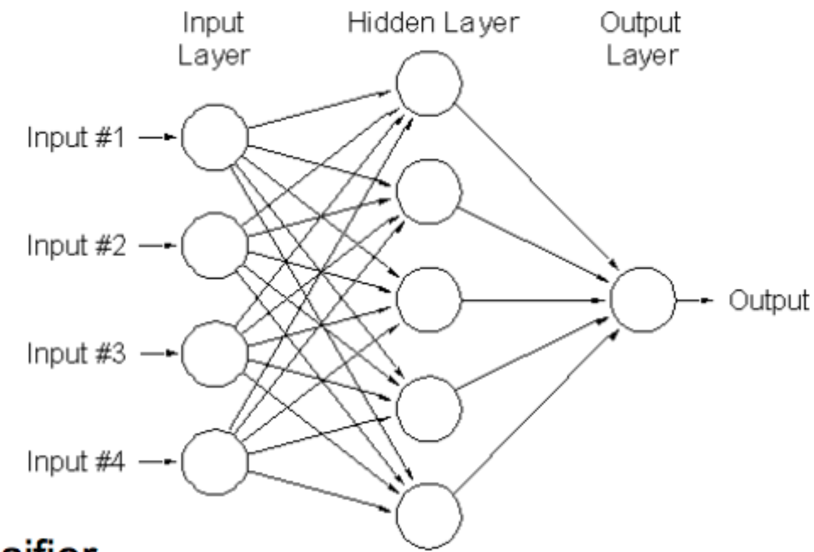


Exercise in Class

Input	Parameters	Output
$X_1=2$	$w_{14}=2$, $w_{46}=1$	$X_6=?$
$X_2=1$	$W_{24}=1$, $w_{56}=0$	
$X_3=0$	$b_4=1$, $b_6=0$	

$$\frac{1}{1+e^{-\left(b_j+\sum_{i=1}^{I} w_{ij}x_i\right)}} = x_j$$

Background: Multi-Layer Neural Networks



- Nonlinear classifier
- **Training:** find network weights **w** to minimize the error between true training labels y_i and estimated labels $f_{\mathbf{w}}(\mathbf{x}_i)$:

$$E(\mathbf{w}) = \sum_{i=1}^{N} \left(y_i - f_{\mathbf{w}}(\mathbf{x}_i)\right)^2$$

- Minimization can be done by gradient descent provided *f* is differentiable
  - This training method is called **back-propagation**

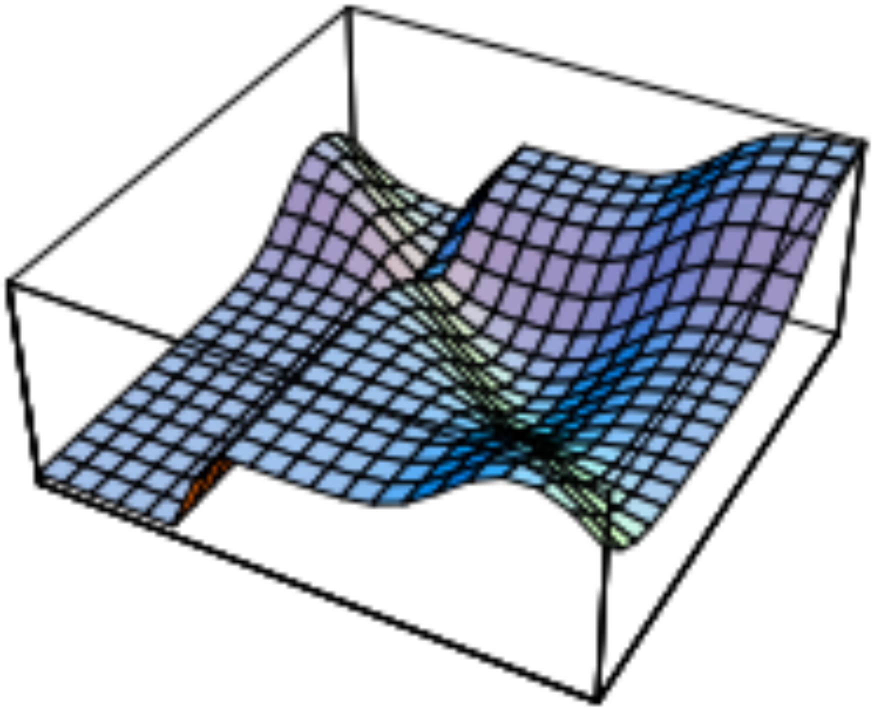
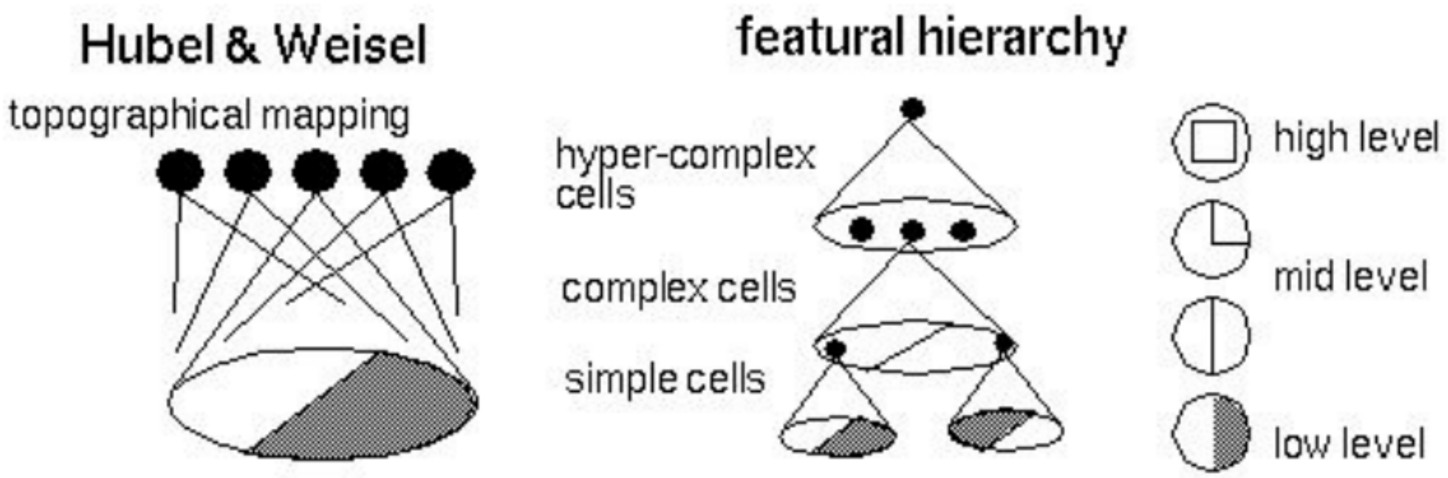


Fig. 7.5. A local minimum of the error function

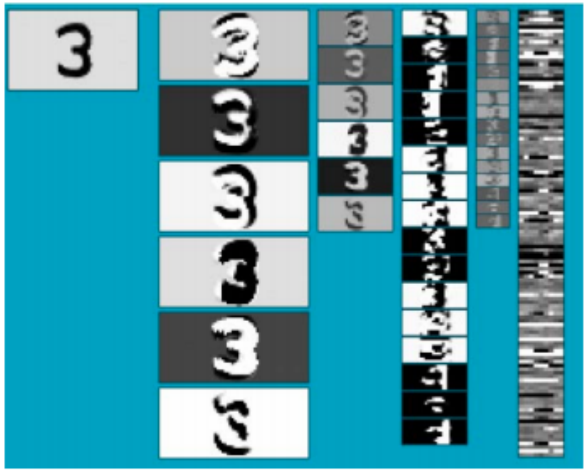
Hubel/Wiesel Architecture

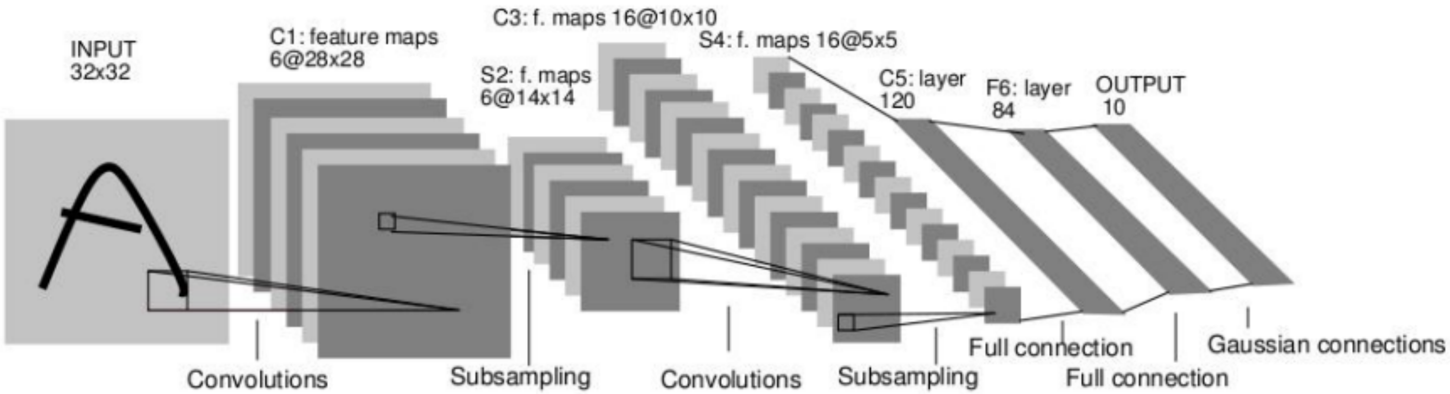
- D. Hubel and T. Wiesel (1959, 1962, Nobel Prize 1981)
  - Visual cortex consists of a hierarchy of *simple*, *complex*, and *hyper-complex* cells



Convolutional Neural Networks (CNN, Convnet)

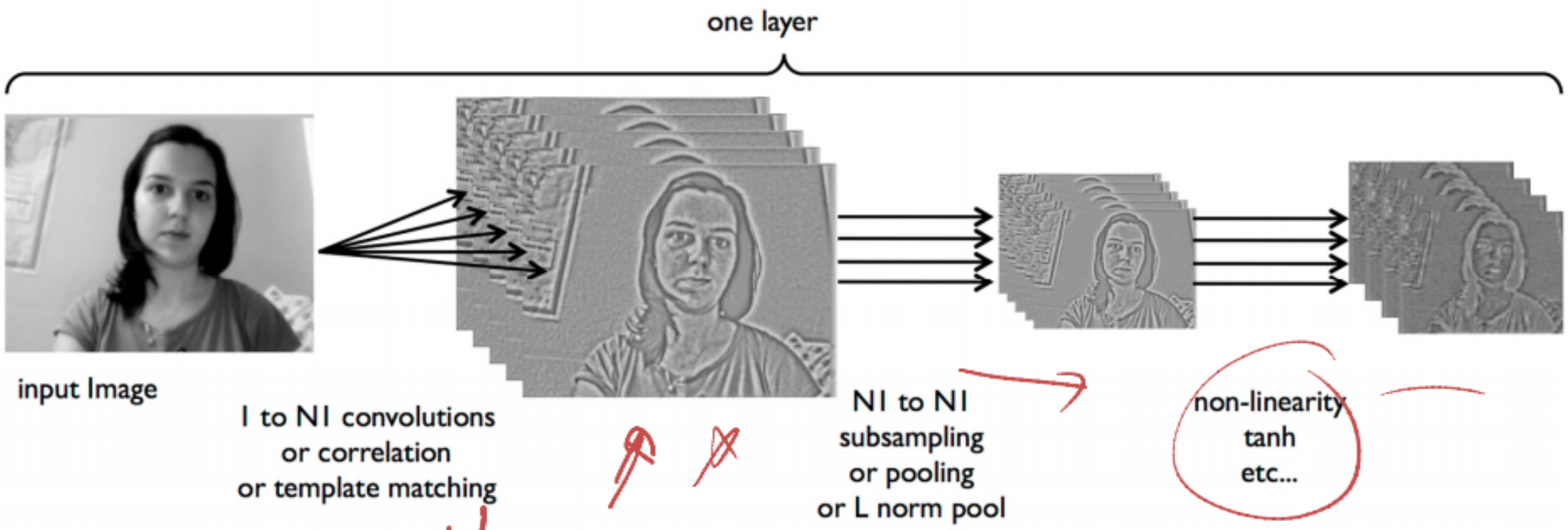
- Neural network with specialized connectivity structure
- Stack multiple stages of feature extractors
- Higher stages compute more global, more invariant features
- Classification layer at the end





Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86(11): 2278–2324, 1998.

Convnets (Fukushima, LeCun, Hinton)



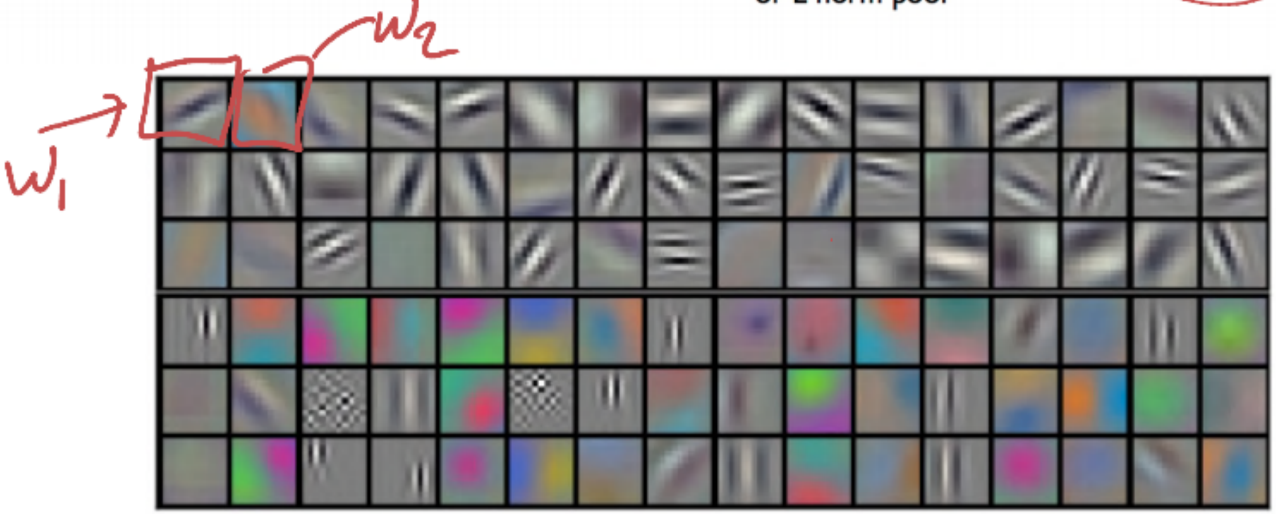
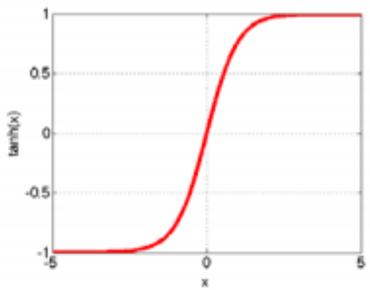


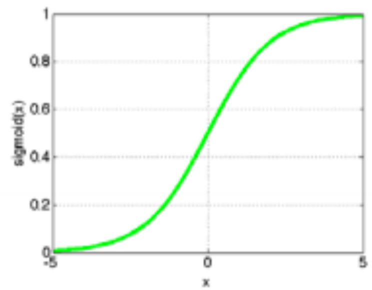
Image max-pooling layer

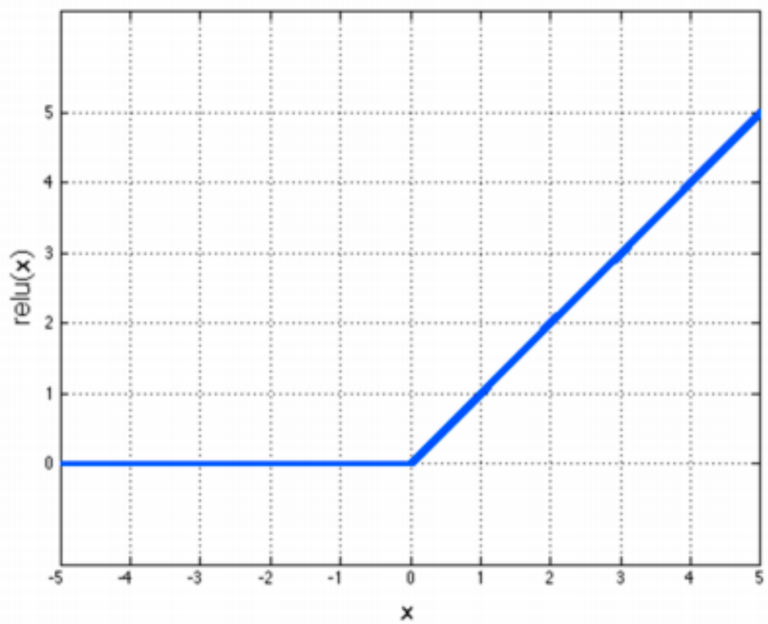
$$\mathbf{y}_{i',j'} = \max_{ij \in \Omega(i'j')} \mathbf{x}_{ij}$$

2. Non-Linearity

- Per-element (independent)
- Options:
  - Tanh
  - Sigmoid: 1/(1+exp(-x))
  - Rectified linear unit (ReLU)
    - Simplifies backpropagation
    - Makes learning faster
    - Avoids saturation issues
      → Preferred option

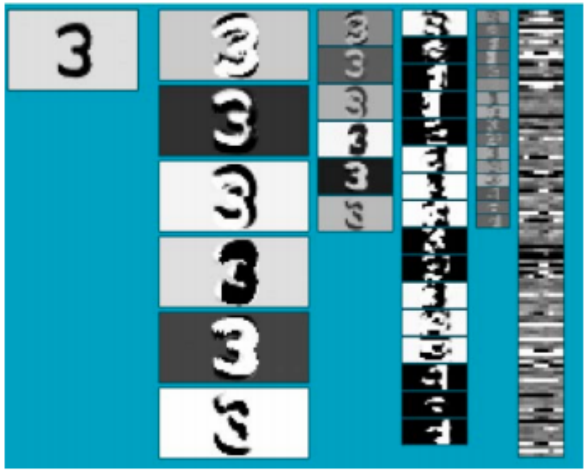


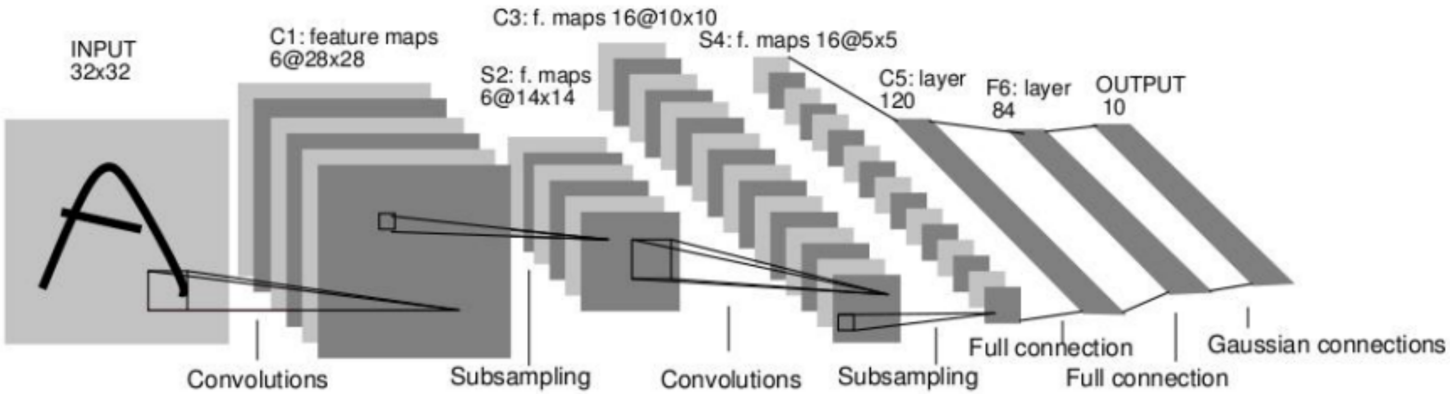




Convolutional Neural Networks (CNN, Convnet)

- Neural network with specialized connectivity structure
- Stack multiple stages of feature extractors
- Higher stages compute more global, more invariant features
- Classification layer at the end





Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86(11): 2278–2324, 1998.

Try it!

- Call cnet_tall (from the cnn Toolbox) and draw digits 0-9 and see if the algorithm can recognize them.
- Try to write as badly as possible (rotation, small-large, noise) and see when the system breaks.

- Classify the first test image from the data base.

```
tmp = load('/Users/hendrik/Library/Matlab/toolboxes/cnn0.83/cnet.mat');
cnet =   tmp.sinet;
[I,labels,I_test,true_labs] = readMNIST(1);
im=preproc_image(I_test{1}); %preprossing of image
[out cnet]=sim(cnet,im'); % classification
[dumm mxidx]=max(out); % getting the largest output node
res=mxidx-1; % started from 0 so add 1
true_labs==res % check if true
```

Issues

- Architecture
- Initialization with small values

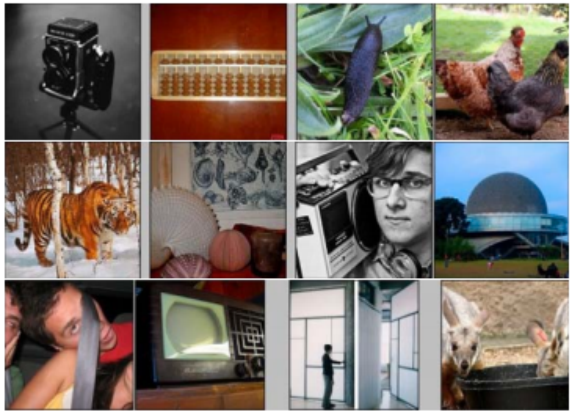
Large Data Bases

- 1.2 Million images of IMAGENET Large Scale Visual Recognition Challenge 2015 (ILSVRC2015)
- Data base can be extended by artificial data
  - e.g. using e.g. 3-D model and rotating it
- MNIST: 70 000 hand-written digits

Convnet Successes

- Handwritten text/digits
  - MNIST (0.17% error [Ciresan et al. 2011])
  - Arabic & Chinese [Ciresan et al. 2012]
- Simpler recognition benchmarks
  - CIFAR-10 (9.3% error [Wan et al. 2013])
  - Traffic sign recognition
    - 0.56% error vs 1.16% for humans [Ciresan et al. 2011]
- But until recently, less good at more complex datasets
  - Caltech-101/256 (few training examples)

ImageNet Challenge 2012

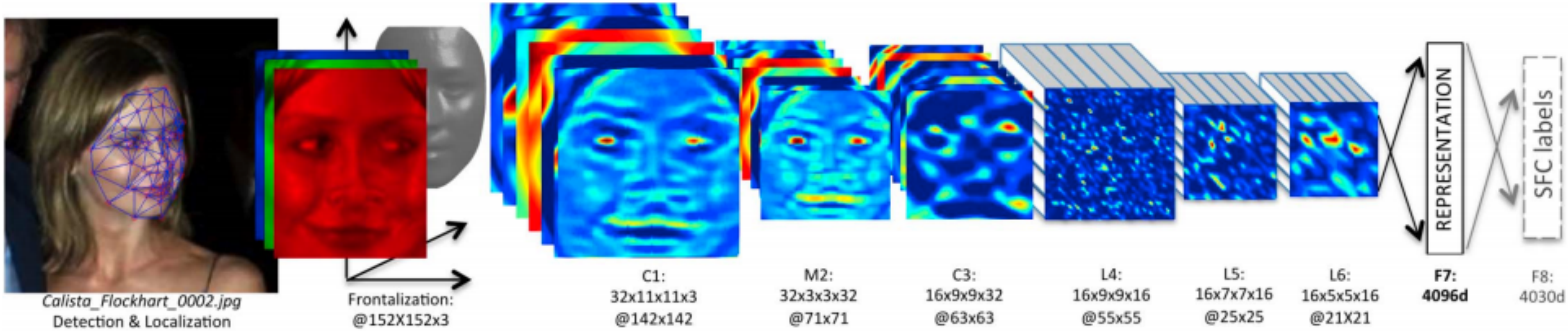


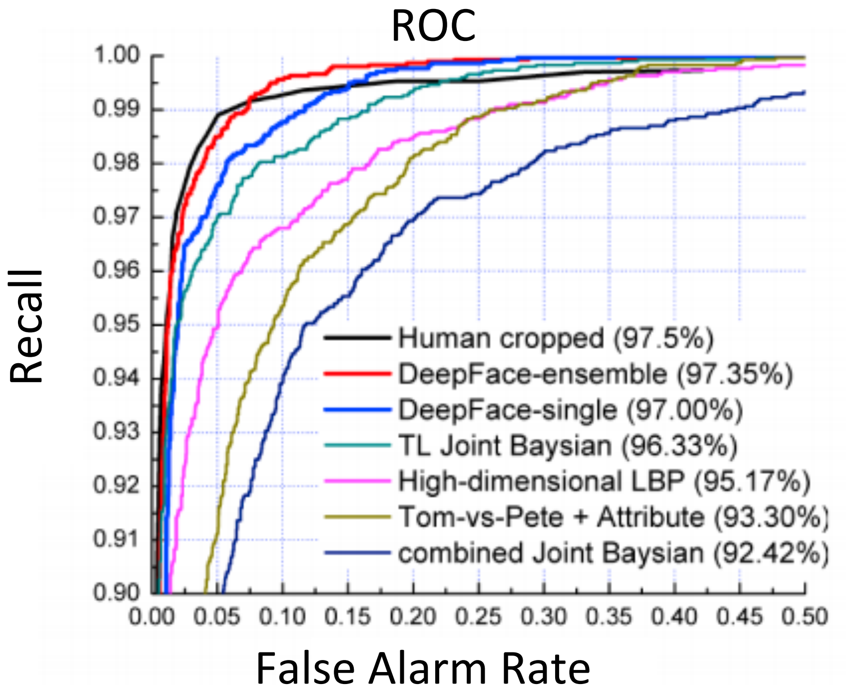
[Deng et al. CVPR 2009]

- ~14 million labeled images, 20k classes
- Images gathered from Internet
- Human labels via Amazon Turk
- Challenge: 1.2 million training images, 1000 classes

A. Krizhevsky, I. Sutskever, and G. Hinton, [ImageNet Classification with Deep Convolutional Neural Networks](), NIPS 2012

CNN features for face verification





Y. Taigman, M. Yang, M. Ranzato, L. Wolf, DeepFace: Closing the Gap to Human-Level Performance in Face Verification, CVPR 2014, to appear.

LeNet Demos

- http://yann.lecun.com/exdb/lenet/index.html

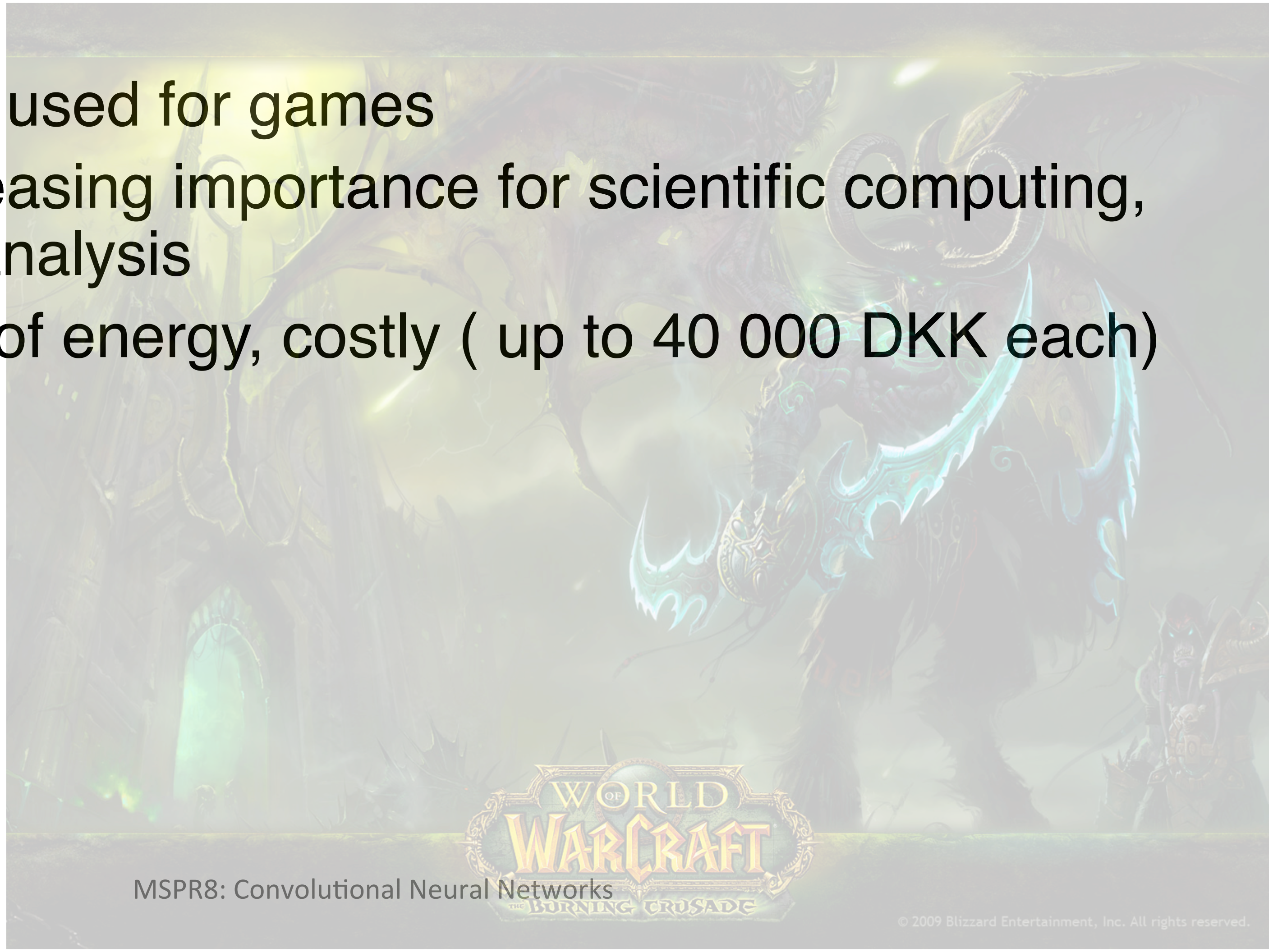
Large Computation Power through Highly Parallel Processing

- Clusters of computers
-

Andrew Ng: Google Brain https://www.youtube.com/watch?v=W15K9PegQt0

Large Computation Power through Highly Parallel Processing

- GPUs (Graphical Processing Units, NVIDIA, Intel)
  - Originally used for games
  - Now increasing importance for scientific computing, big data analysis
  - Use a lot of energy, costly (up to 40 000 DKK each)



Large Computation Power through Highly Parallel Processing

- Technology:
  - Hadoop
    - https://hadoop.apache.org/
  - CUDA modelling framework
    - http://www.nvidia.com/object/cuda_home_new.html
  - Deep learning framework:
    - http://caffe.berkeleyvision.org/

References

- Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86(11): 2278–2324, 1998.
- Nando de Freitas: Lecture on Convolutional Neural Networks
  - Video: http://www.youtube.com/watch?v=bEUX_56Lojc
  - Slides: http://www.cs.ox.ac.uk/people/nando.defreitas/machinelearning/lecture9.pdf
- Svetlana Lazebnik Illinois, Rob Fergus (NYU)
  - http://web.engr.illinois.edu/~slazebni/spring14/lec24_cnn.pdf
- Hinton: Convolutional nets for digit recognition
  - http://www.youtube.com/watch?v=6oD3t6u5EPs