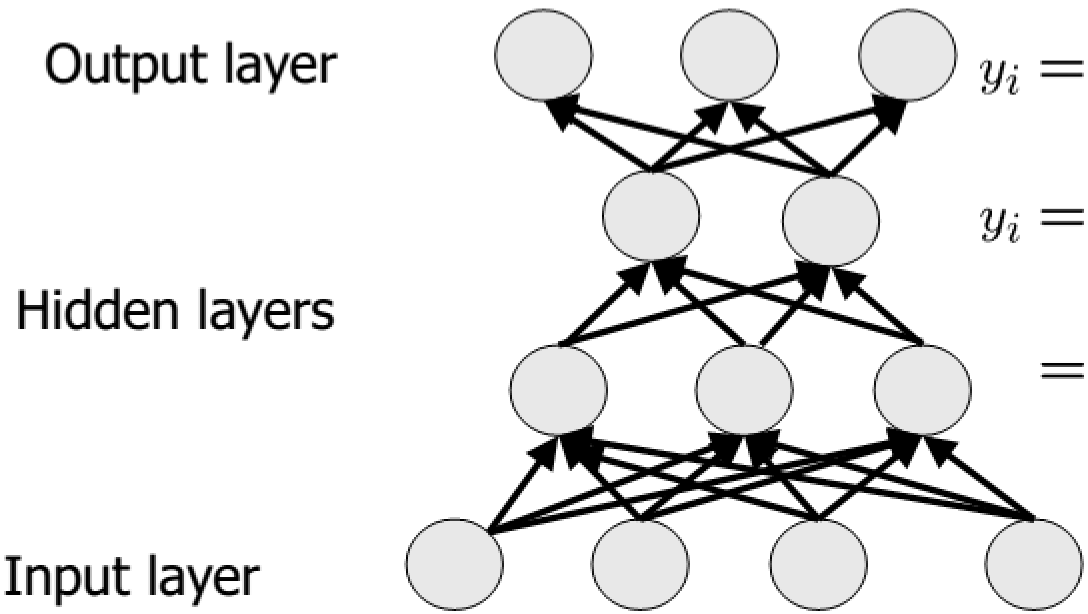
Deep networks for classification

From NN to DL

- Classical NN



$$y_i = \tanh\left(\sum_j W_{ij} x_j + W_{i0}\right)$$

$$y_i = \sigma\left(\sum_j W_{ij} x_j + W_{i0}\right)$$

$$= \frac{1}{1 + \exp[-(\sum_j W_{ij} x_j + W_{i0})]}$$

- Open questions: how many hidden layer?

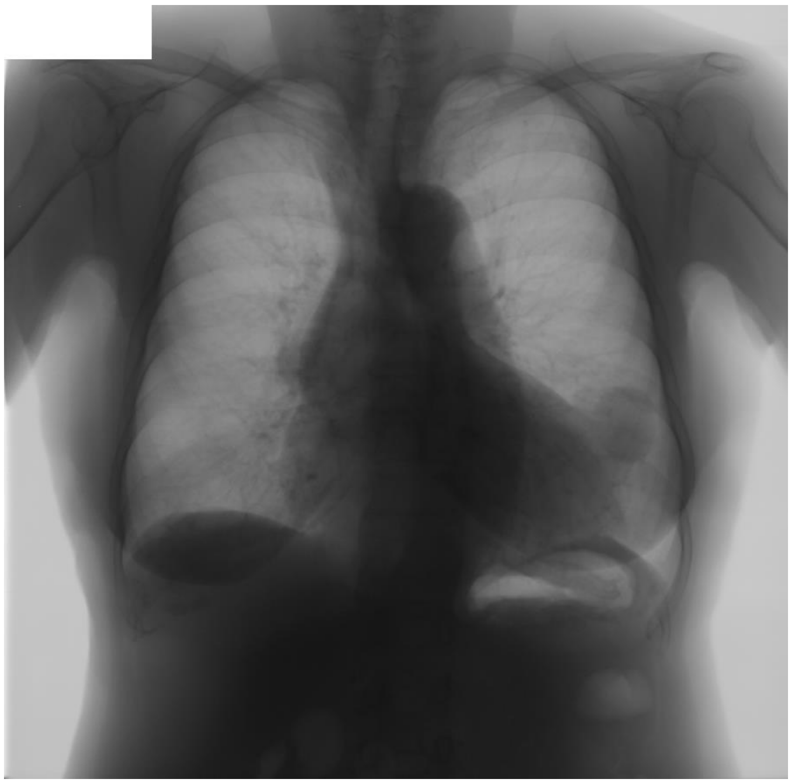
MLP with many hidden layers

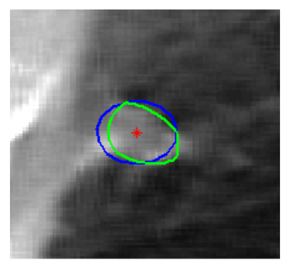
- One hiden layer is enough for the universal approximation property, but there may be advantageous if more hidden layers are used.
  - More complex mapping with smaller number of neurons in the hidden layers
  - Different types of hidden layers can be applied
- Drawbacks
  - BP training is slow
  - Too many free parameters, too large degree of freedom
  - Intensive computation

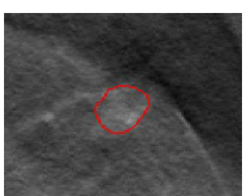
Classification

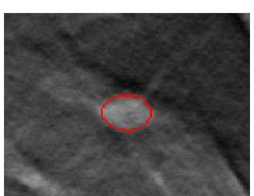
- Input data collection from the cases to be classified
- Definition of descriptive parameters
  - Examples (industrial problem, diagnostic problem,...)
  - Image clasification
    - ROI selection,
    - features of the ROIs, construction of feature vectors
    - Construct a clasifier (MLP, Basis function network, SVM,...) based on the feature vectors and the corresponding labels
    - Train the classifier

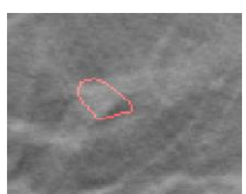
Medical diagnosis as a classification problem

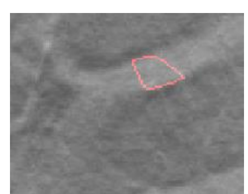


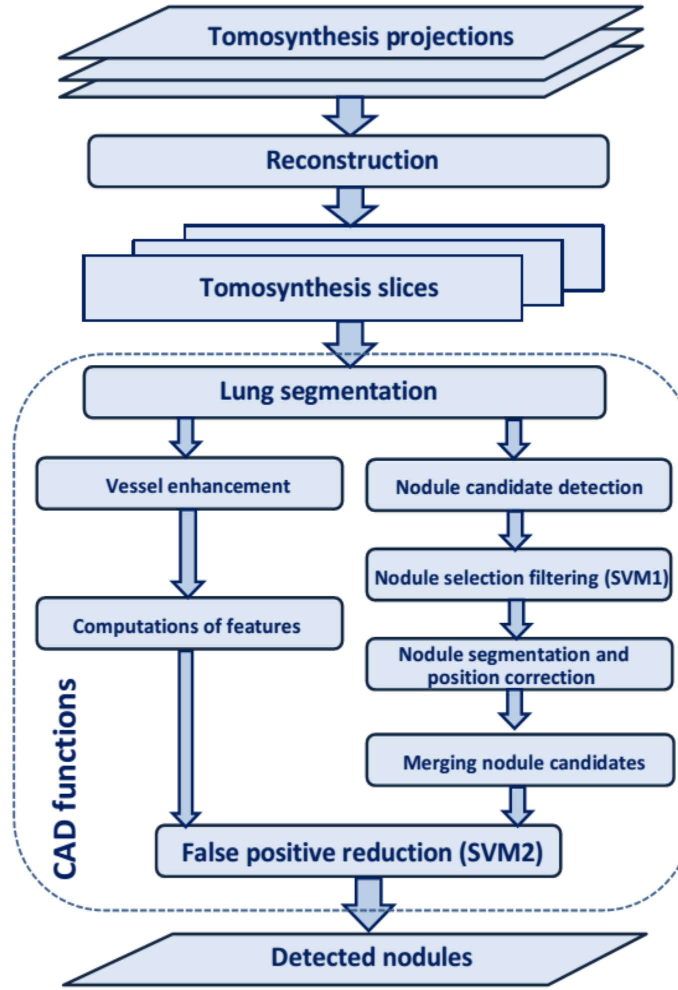




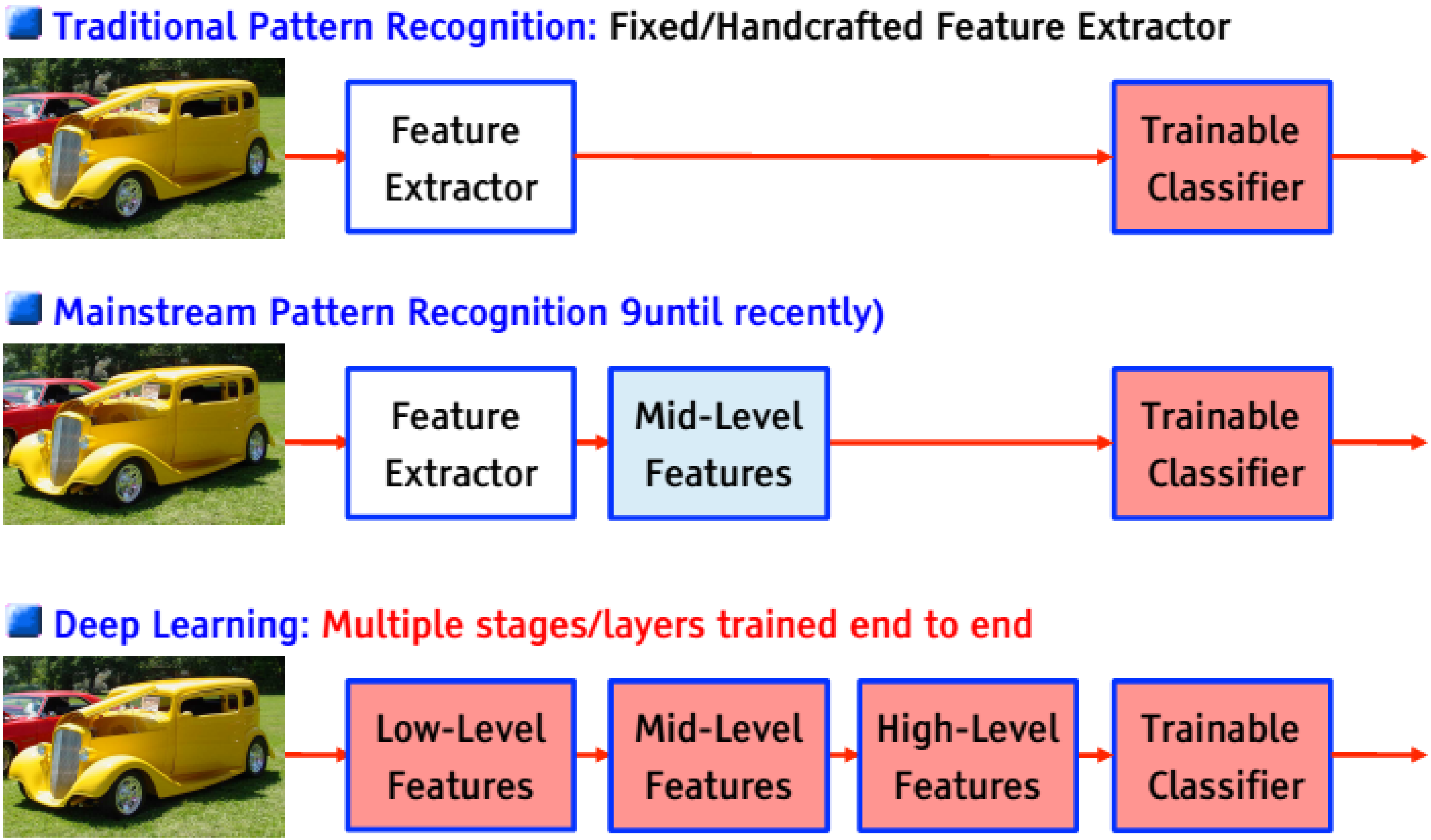








Basic steps of classification



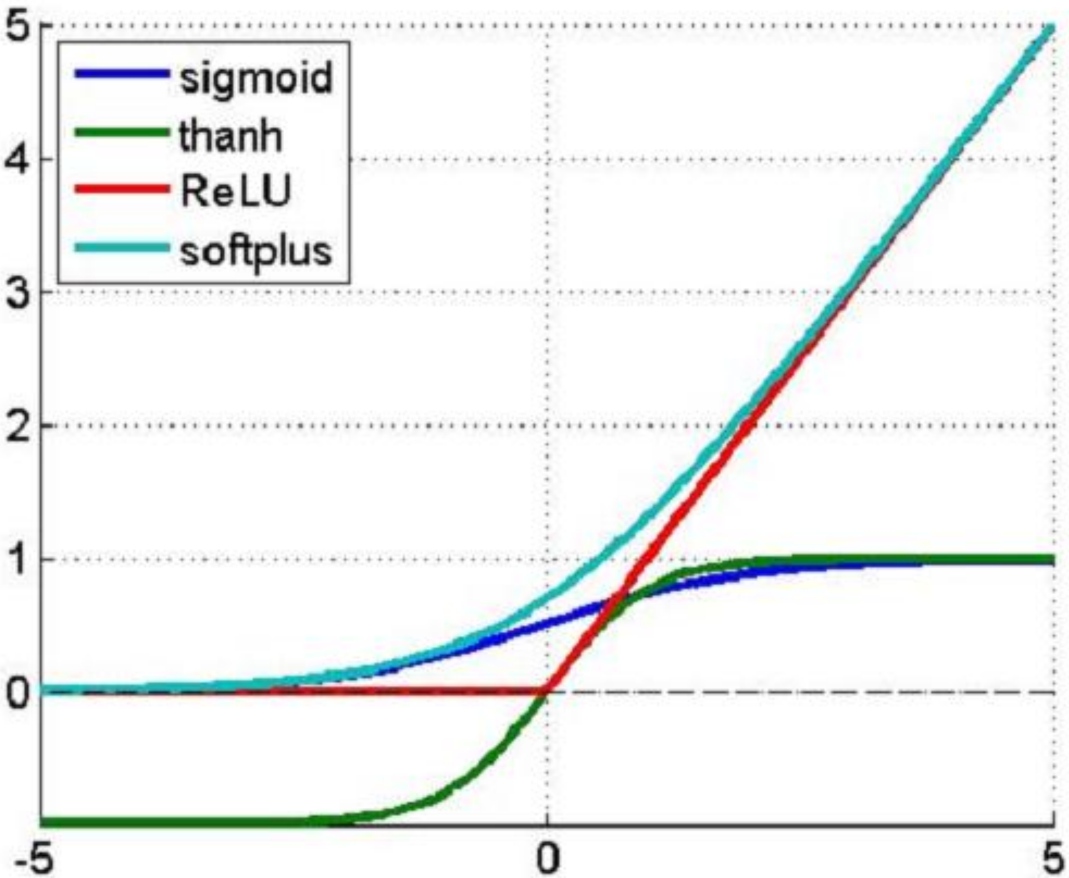
Feature selection

- Dimension reduction to determine the most important features (PCA, KPCA)
- Dimension reduction to determine the most relevant features (PLS, sensitivity analysis)
- Looking for a sparse solution (regularization, ...)

Training

- MLP BP algorithm
  - Drawbacks saturating nonlinear activation function
    - Sigmoidal nonlinearity, the derivatives go to zero ....
    - Calculation of exponential function values
    - Slow and computationally complex algorithm
    - Stick at a local minimum
  - How to improve the architecture for avoiding the drawbacks
    - Change the activation function

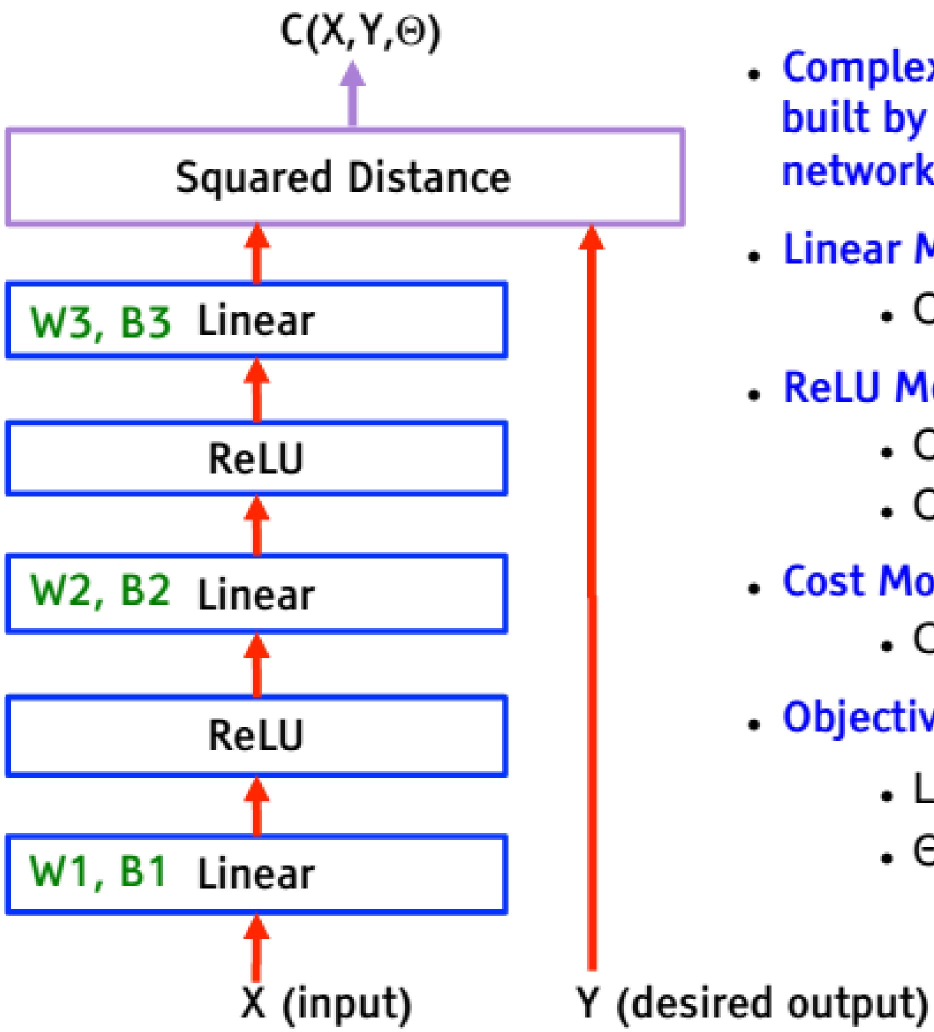
Activation functions



Advantages

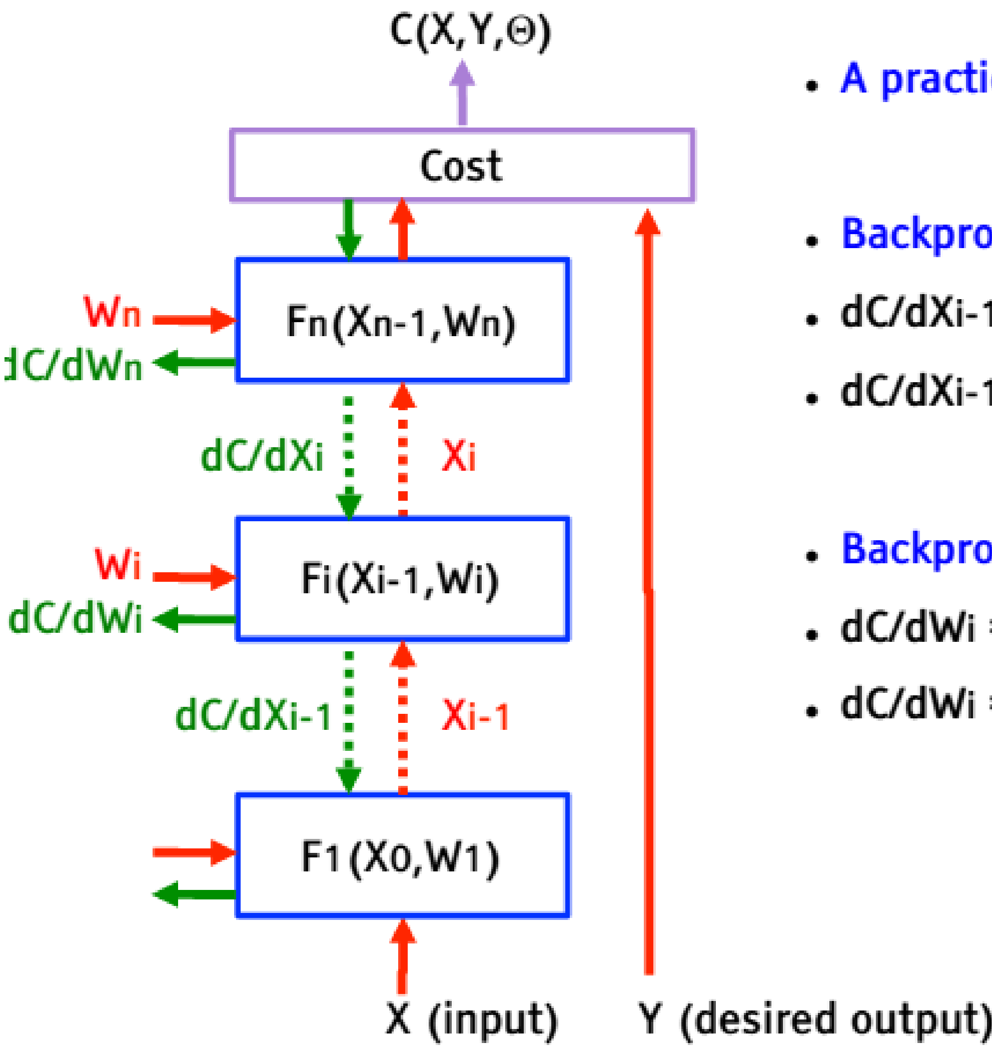
- Easy to calculate
- No saturation
- No required complex derivative
- Univ approximation capability remains
- Efficient gradient-based learning algorithms

A new MLP architecture



- Complex learning machines can be built by assembling modules into networks
- Linear Module
  - Out = W.In+B
- ReLU Module (Rectified Linear Unit)
  - $Out_i = 0$ if $In_i<0$
  - $Out_i = In_i$ otherwise
- Cost Module: Squared Distance
  - $C = ||In1 - In2||^2$
- Objective Function
  - $L(\Theta)=1/p \, \Sigma_k \, C(X^k,Y^k,\Theta)$
  - $\Theta = (W1,B1,W2,B2,W3,B3)$

Training (BP)



- A practical Application of Chain Rule
- Backprop for the state gradients:
- $dC/dX_{i-1} = dC/dX_i \cdot dX_i/dX_{i-1}$
- $dC/dX_{i-1} = dC/dX_i \cdot dF_i(X_{i-1},W_i)/dX_{i-1}$
- Backprop for the weight gradients:
- $dC/dW_i = dC/dX_i \cdot dX_i/dW_i$
- $dC/dW_i = dC/dX_i \cdot dF_i(X_{i-1},W_i)/dW_i$

Training algorithms

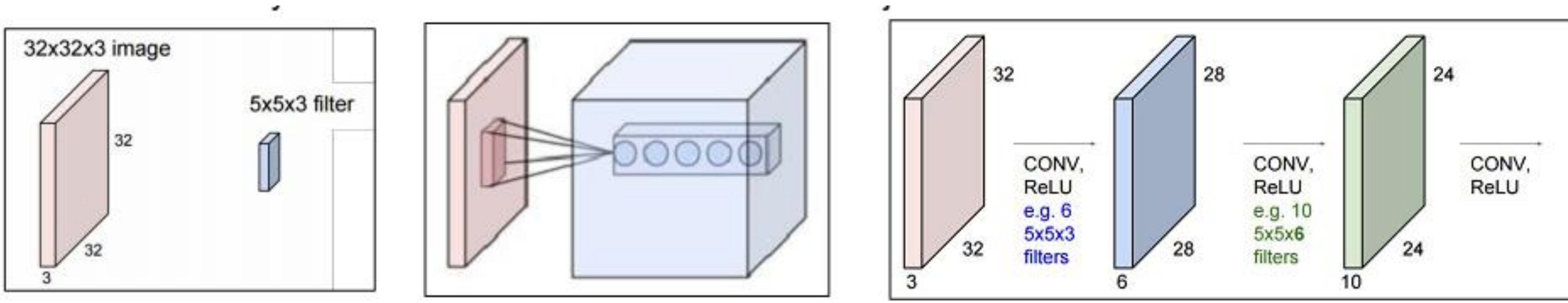
- SGD
- Minibatch
- Different gradient algorithms
- Momentum (Nesterov momentum)

Data set

- Increase the number of labelled data
- Artificially generated samples (augmentations)
  - Shifting
  - Rotating
  - Flip vertically or horizontally
  - ...

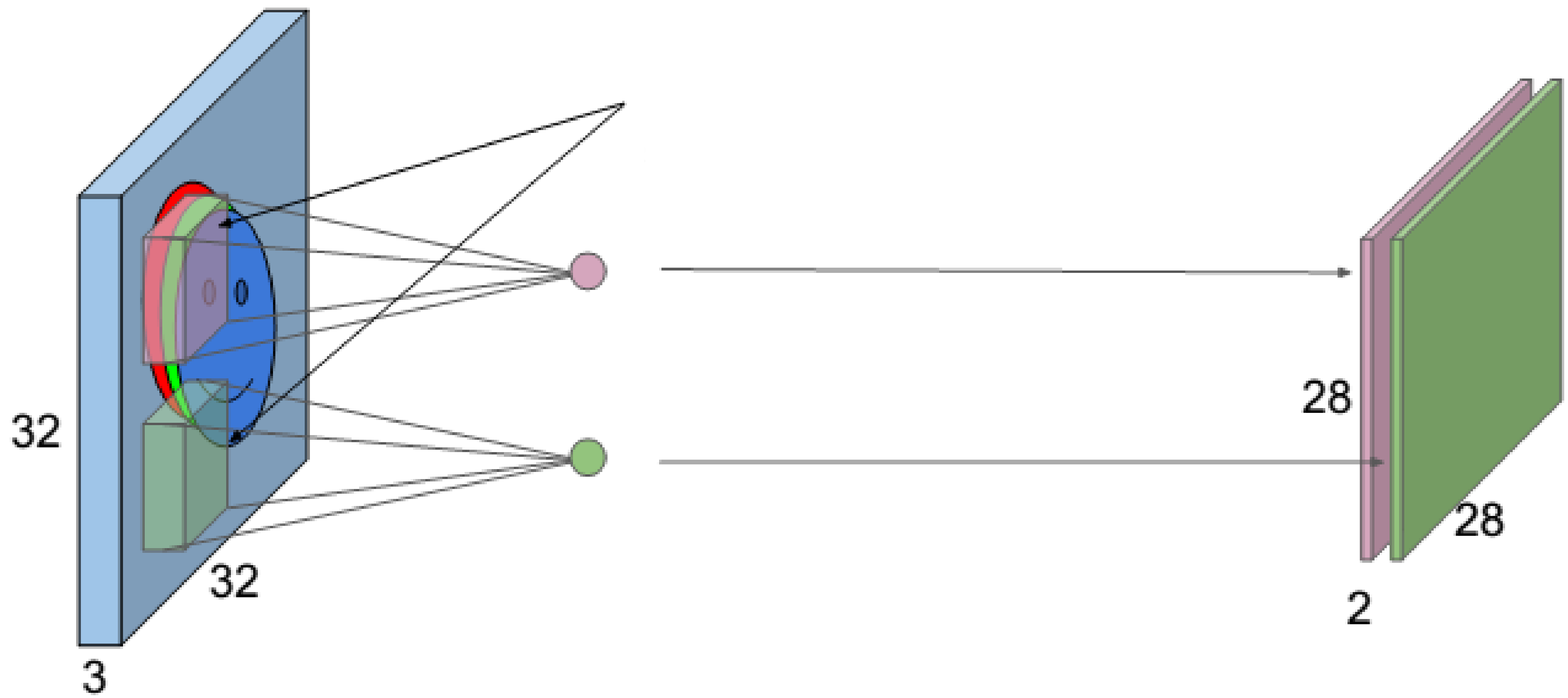
Feature selection

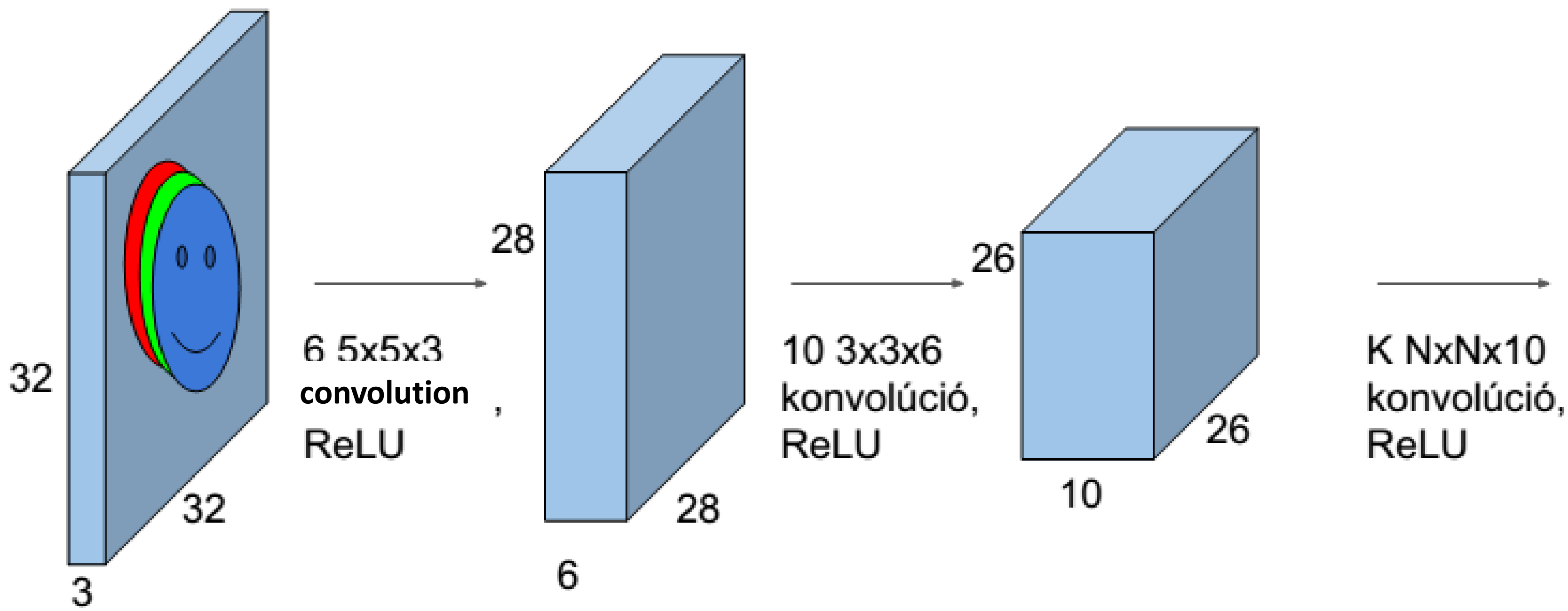
- Introduction of different type-layers



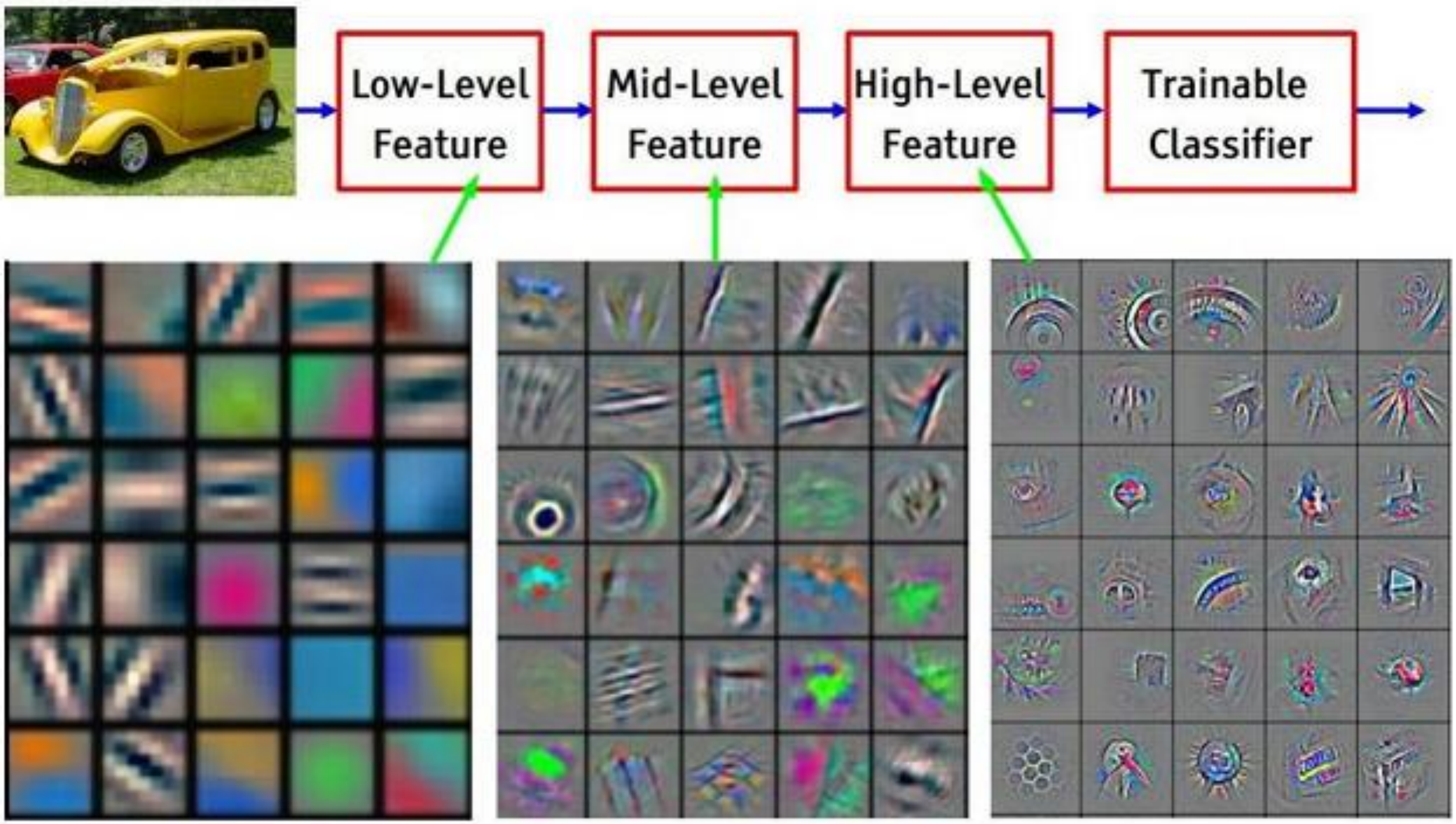
- Feature selection is done by the network itself
  - Filtering (convolution), many convolutional layers
  - Dimension reduction, feature selection

Convolutional layer




32
32
3
6 5x5x3
convolution
,
ReLU
28
28
6
10 3x3x6
konvolúció,
ReLU
26
26
10
K NxNx10
konvolúció,
ReLU

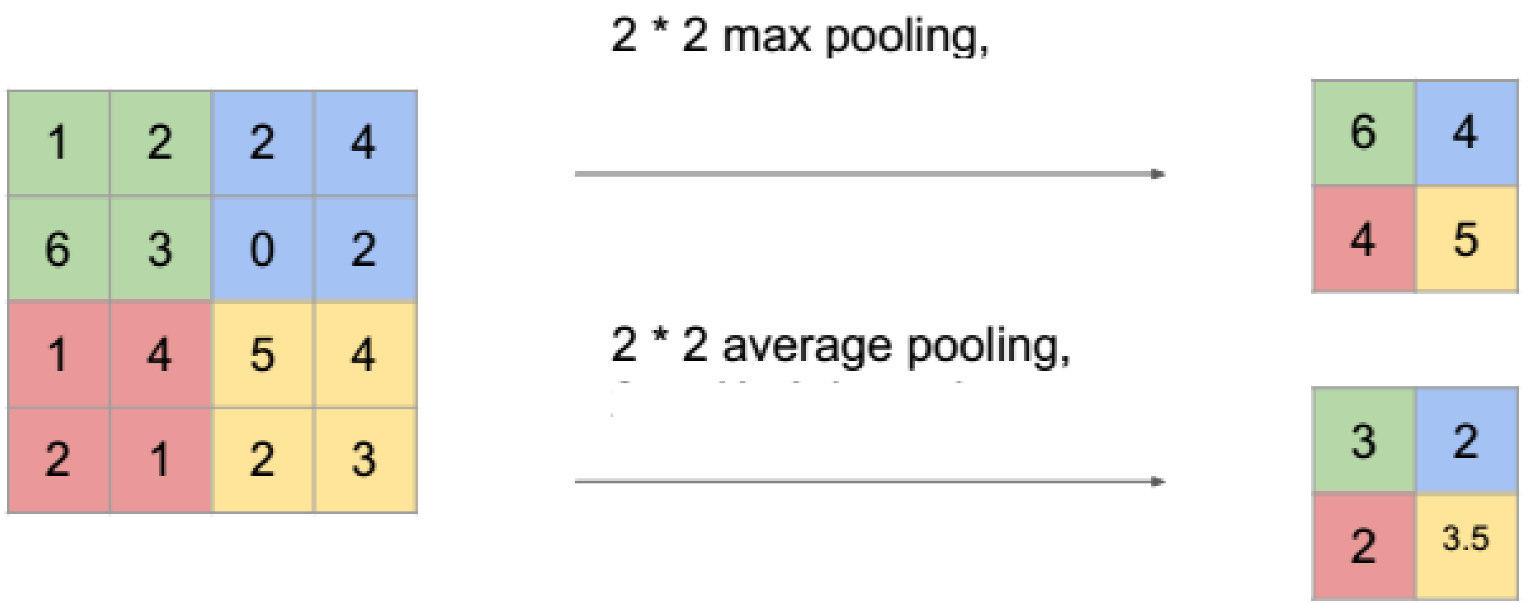
Feature selection



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

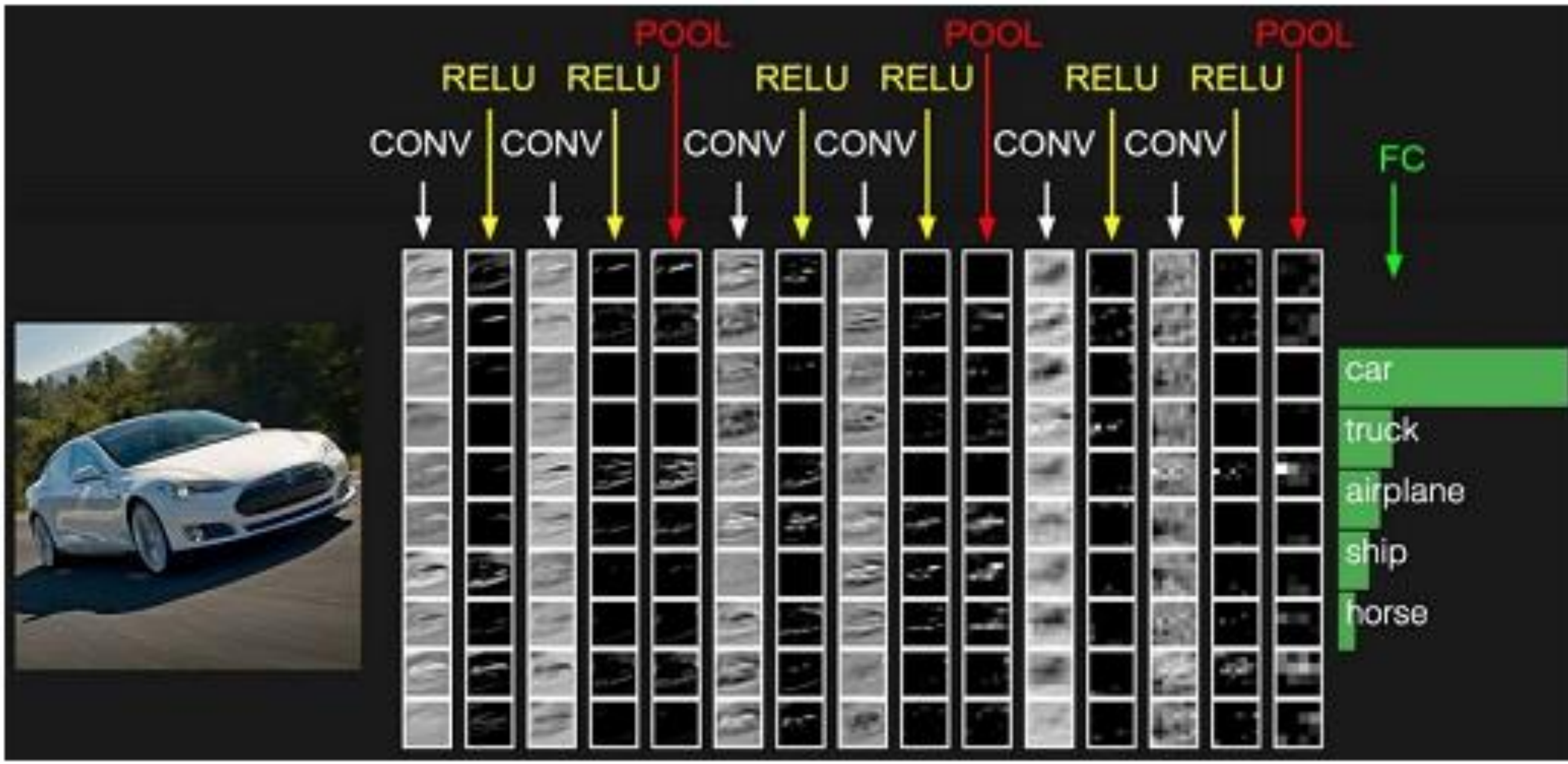
Pooling layer

MAX pooling, average pooling

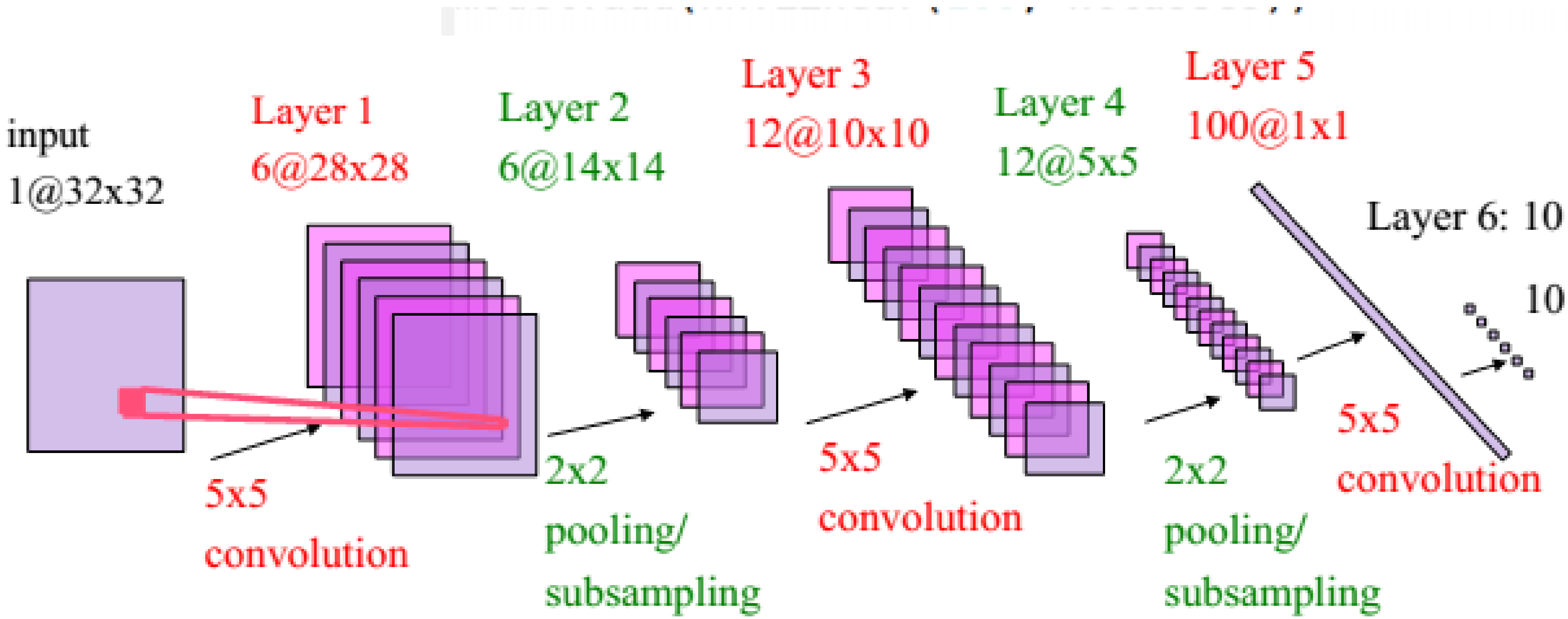


Dimension reduction

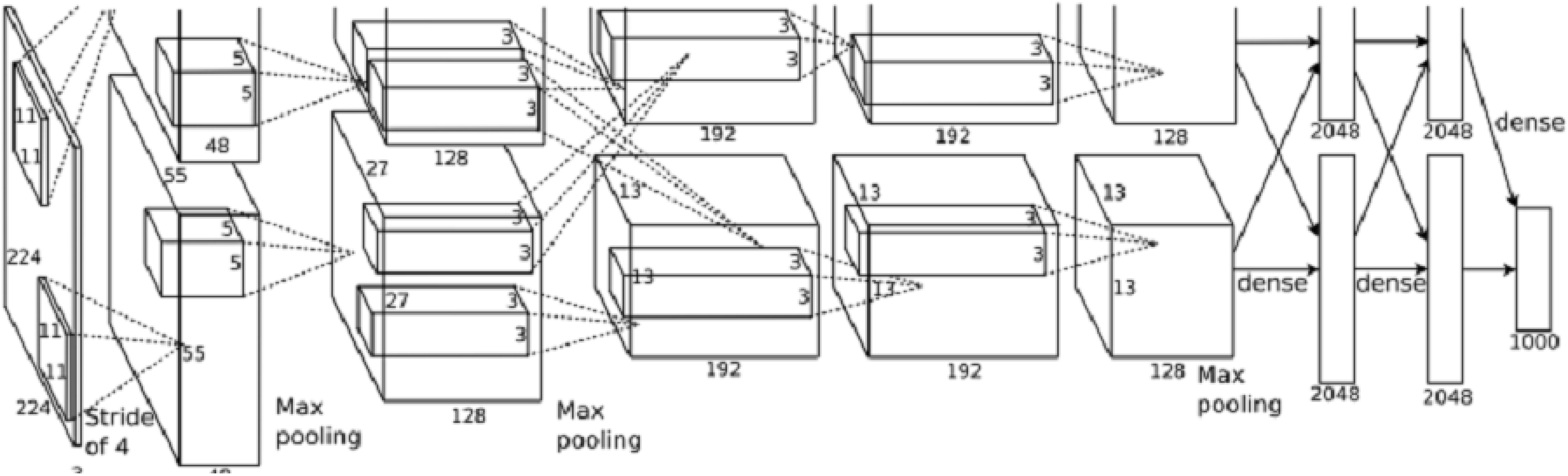
Fully connected layers



A complex network



A complex network

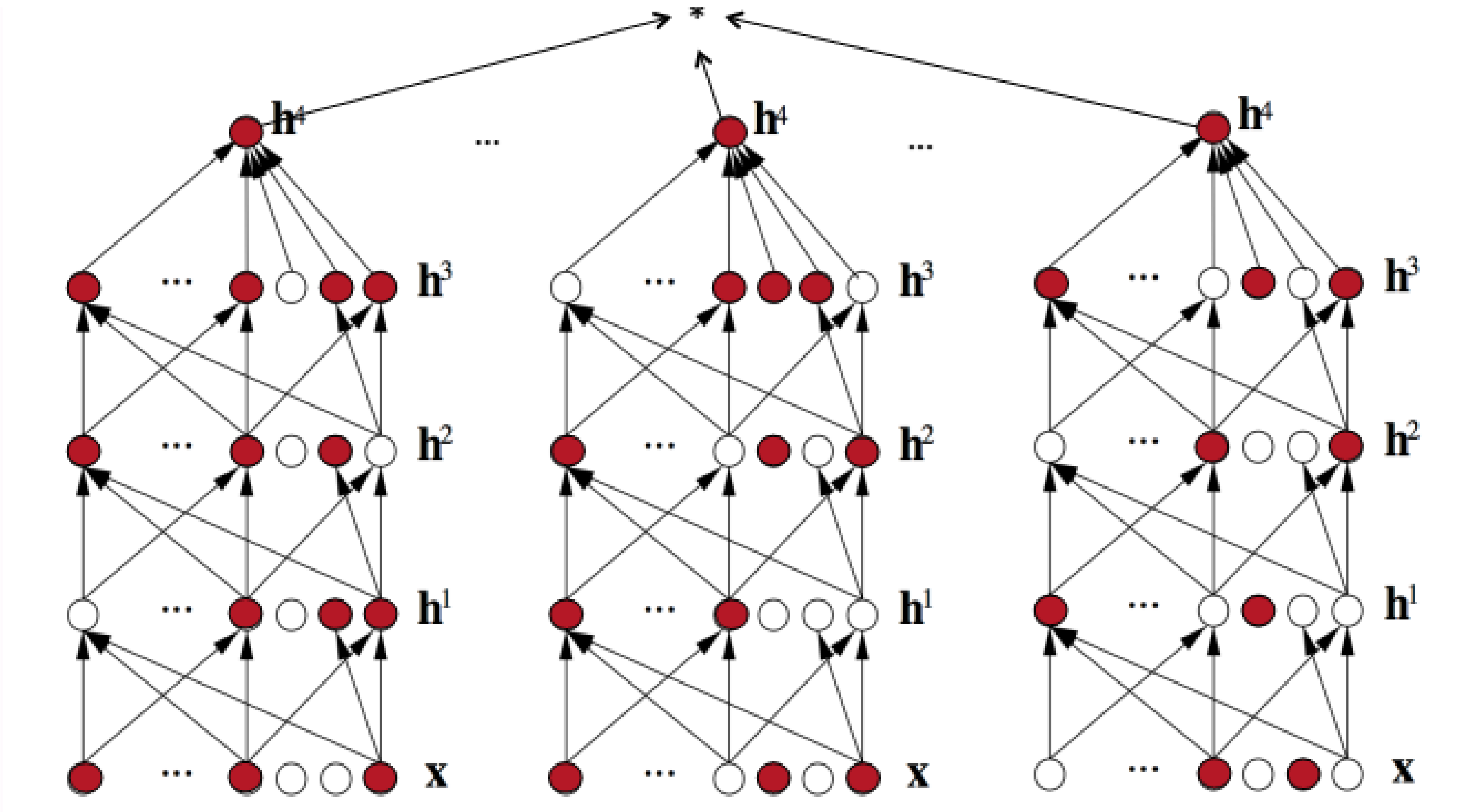


The first convolutional layer filters the 224×224×3 input image with 96 kernels of size 11×11×3 with a stride of 4 pixels (this is the distance between the receptive field centers of neighboring neurons in the kernel map. 224/4=56

The pooling layer: form of non-linear down-sampling. Max-pooling partitions the input image into a set of rectangles and, for each such sub-region, outputs the maximum value

Dropout

Complex neurons (to reduce free parameters)



Dropout: set the output of each hidden neuron to zero w.p. 0.5.

Dropout

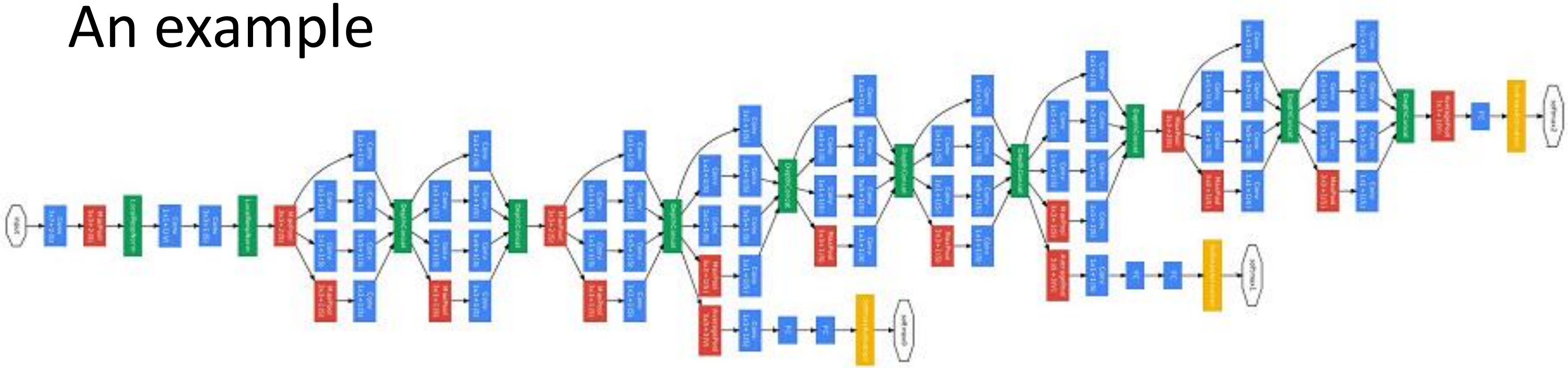
Dropout: set the output of each hidden neuron to zero w.p. 0.5.

- The neurons which are "dropped out" in this way do not contribute to the forward pass and do not participate in backpropagation.
- So every time an input is presented, the neural network samples a different architecture, but all these architectures share weights.
- This technique reduces complex co-adaptations of neurons, since a neuron cannot rely on the presence of particular other neurons.
- It is, therefore, forced to learn more robust features that are useful in conjunction with many different random subsets of the other neurons.
- Without dropout, our network exhibits substantial overfitting.
- Dropout roughly doubles the number of iterations required to converge.

Autoencoders

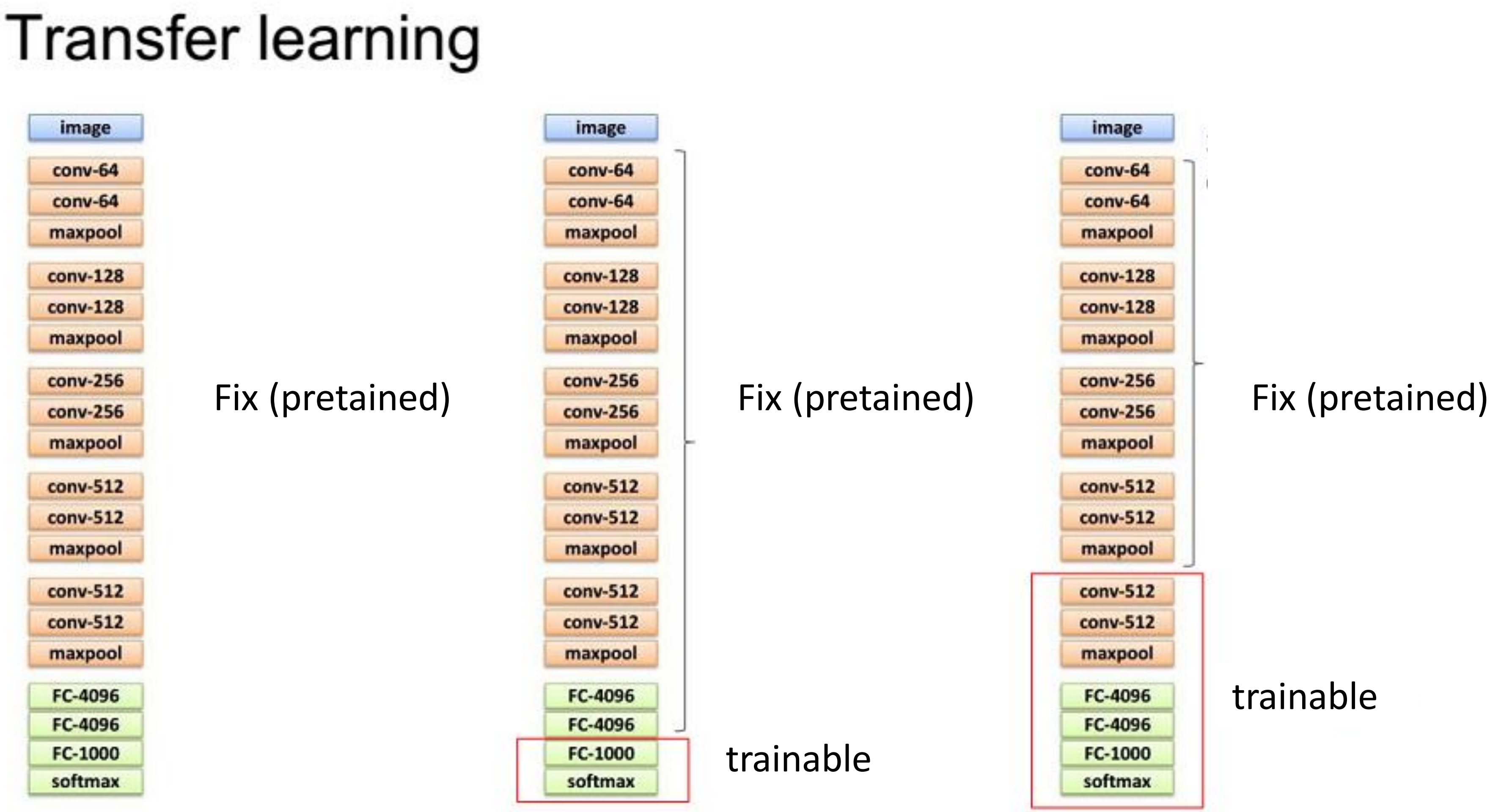
- Feature selection, dimension reduction
- (bottleneck layer)

An example



GoogLeNet *[Szegedy et al., 2014]*

Transfer learning

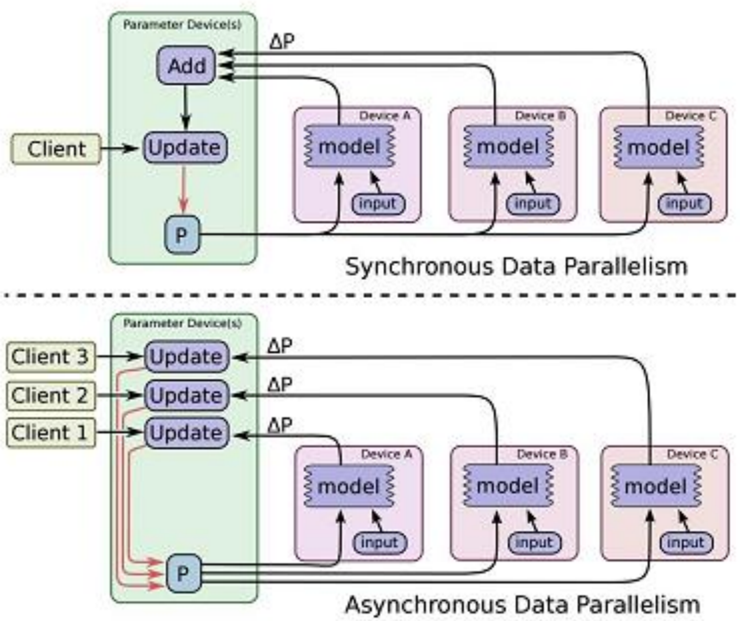


Implementation

TensorFlow

http://download.tensorflow.org/paper/whitepaper2015.pdf

TensorFlow - Multi GPU



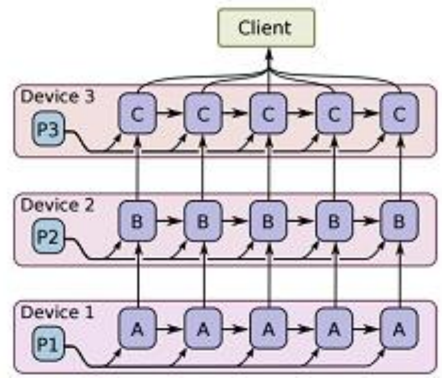


Figure 8: Model parallel training

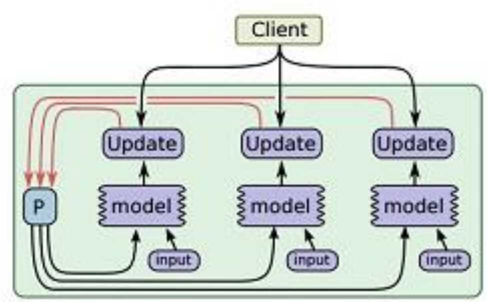
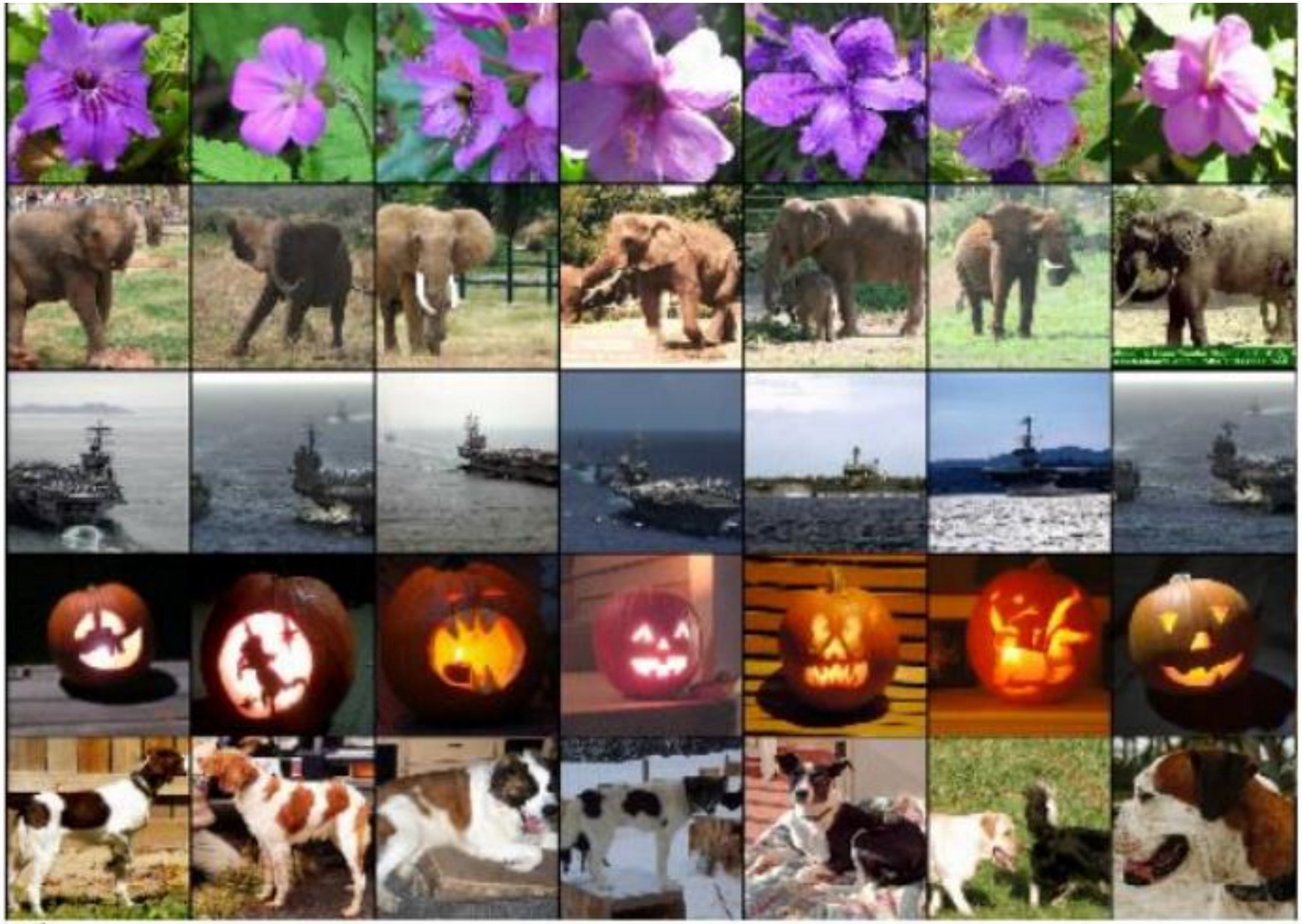


Figure 9: Concurrent steps



Test column

six training images that produce feature vectors in the last hidden layer with the smallest Euclidean distance from the feature vector for the test image.

Implementation

Models:

GoogleNet: CNN model finetuned on the Extended Salient Object Subitizing dataset (~11K images) and synthetic images. This model significantly improves over our previous models. **Recommended**.

AlexNet: CNN model finetuned on our initial Salient Object Subitizing dataset (~5500 images). The architecture is the same as the Caffe reference network.

VGG16: CNN model finetuned on our initial Salient Object Subitizing dataset (~5500 images).

Many further details can be found in http://deeplearning.net/

Some figures of this slide set was obtained from:

- Deep Learning NIPS'2015 Tutorial, Geoff Hinton, Yoshua Bengio & Yann LeCun
- Introduction to Machine Learning CMU-10701 Deep Learning