Neural net applications for images

Victor Kitov

v.v.kitov@yandex.ru

Yandex
School of Data Analysis.

Convolutional neural networks

- Convolutional neural network:
  - Used for image analysis
  - Consists of a set of convolutional layer / sub-sampling (pooling) layer pairs and several fully connected layers at the end.

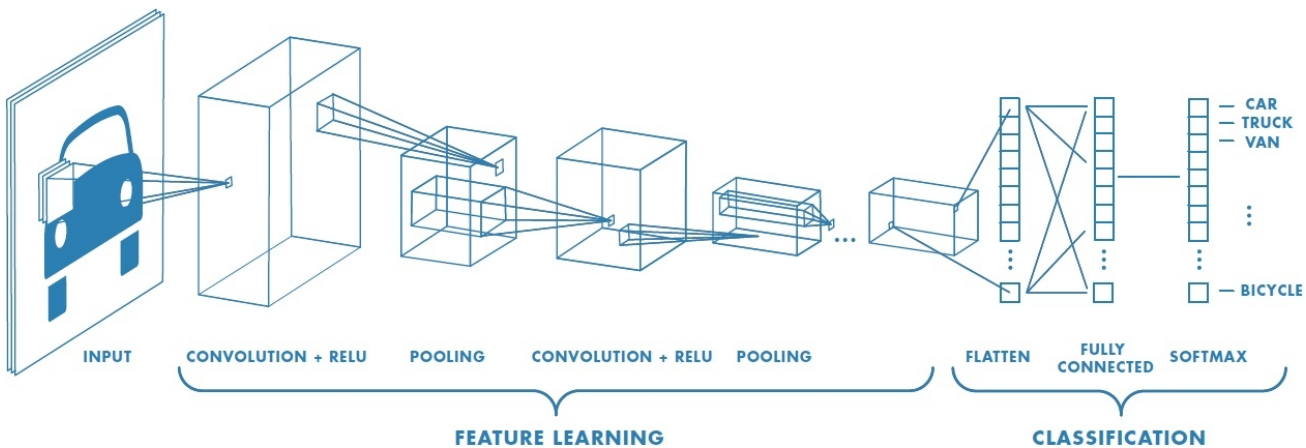
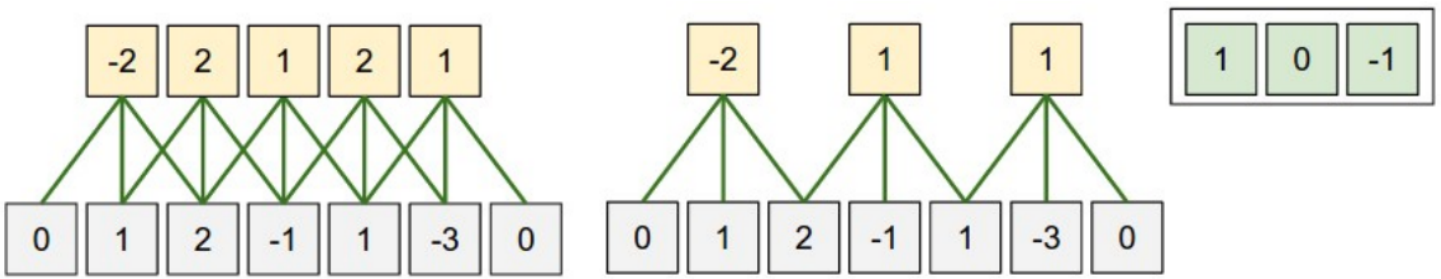


Table of Contents

1-D Convolution operation

1-D convolution [$W = 5$, $K = 3$, zero-padded with $\mathrm{P}{=}1$, $S = 1$ (left) and $S = 2$ (right)]



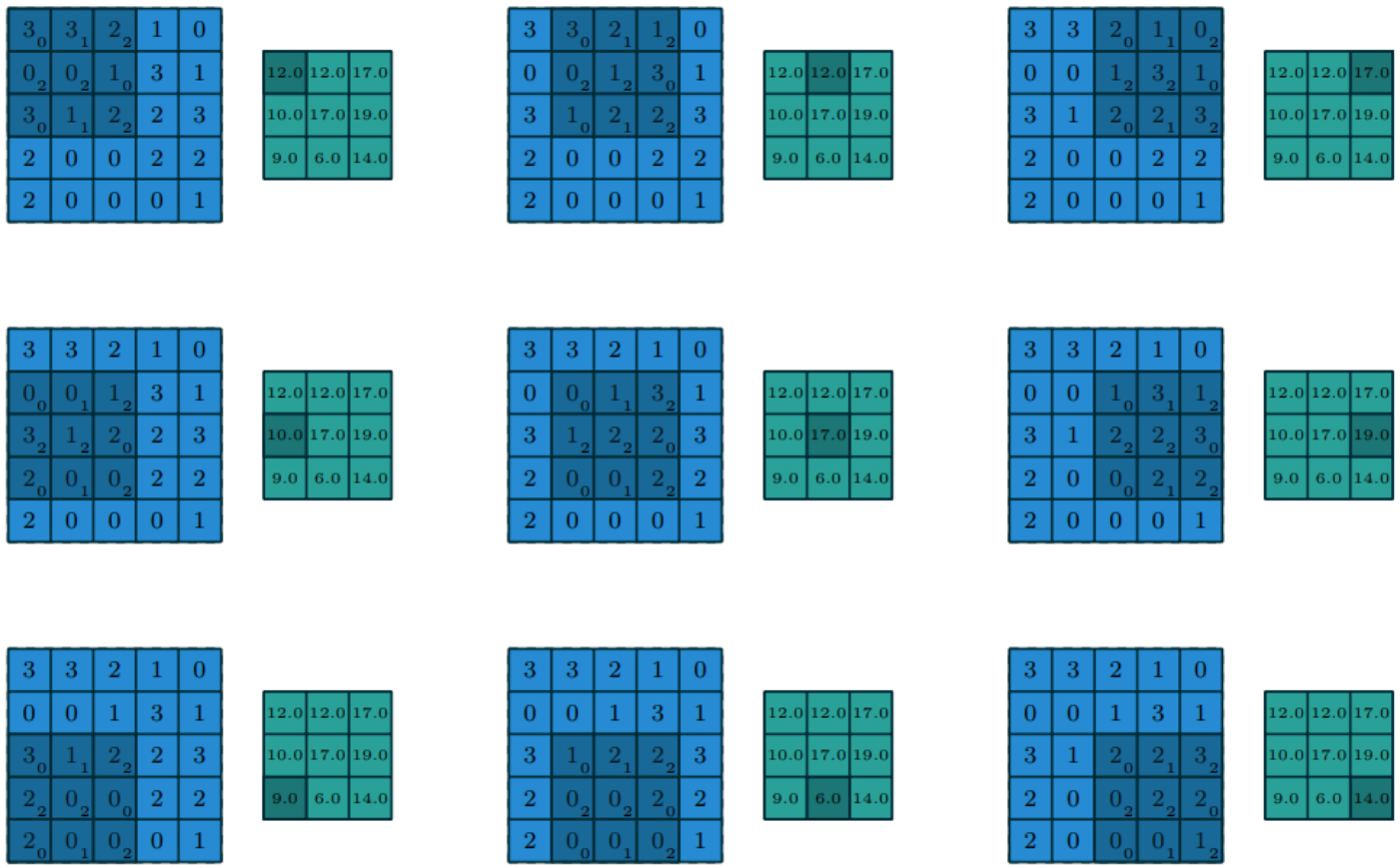
Parameters[1]:

- W - length of input
- K - kernel size
- P - amount of padding
- Type of padding (zero, extension, mirror)
- S - stride (offset of kernel)
- D - dilation (offset inside kernel)

[1] Depending on these parameters, what would be the size of output layer?

Convolution[2]

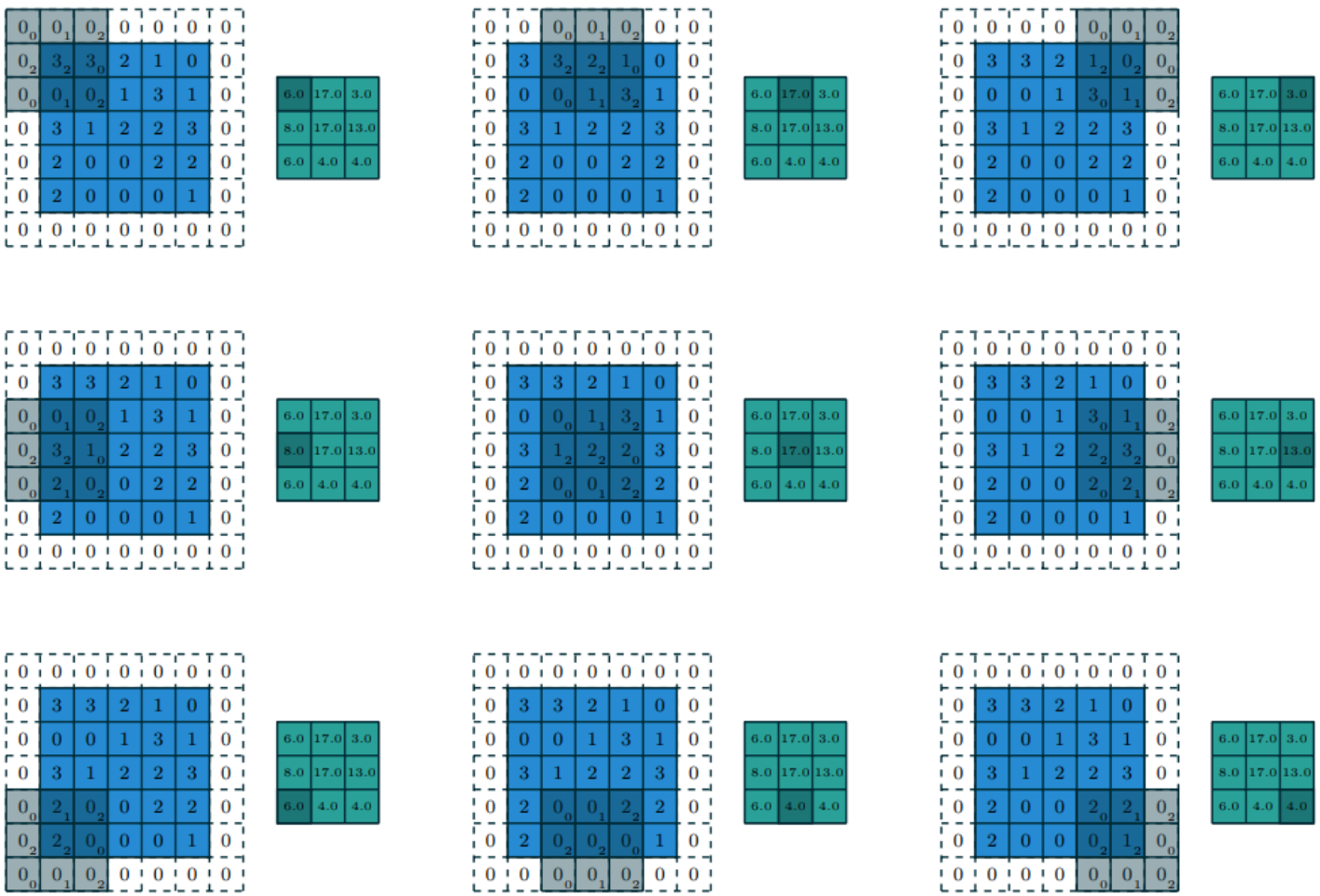
Single layer convolution:



[2]Illustrations from Dumoulin et al. 2018.

Convolution

Convolution with stride and zero-padding:

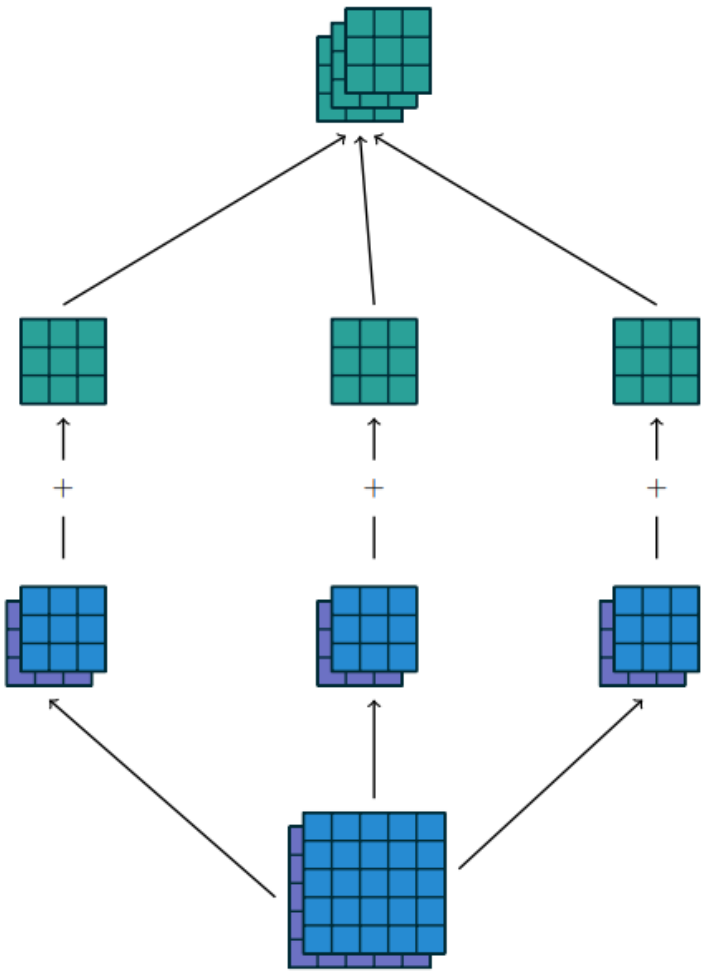


Padding

- Stride: to decrease dimensionality.
- Padding: to increase dimensionality.
- Padding types:
  - zero padding
  - same padding
  - mirror padding

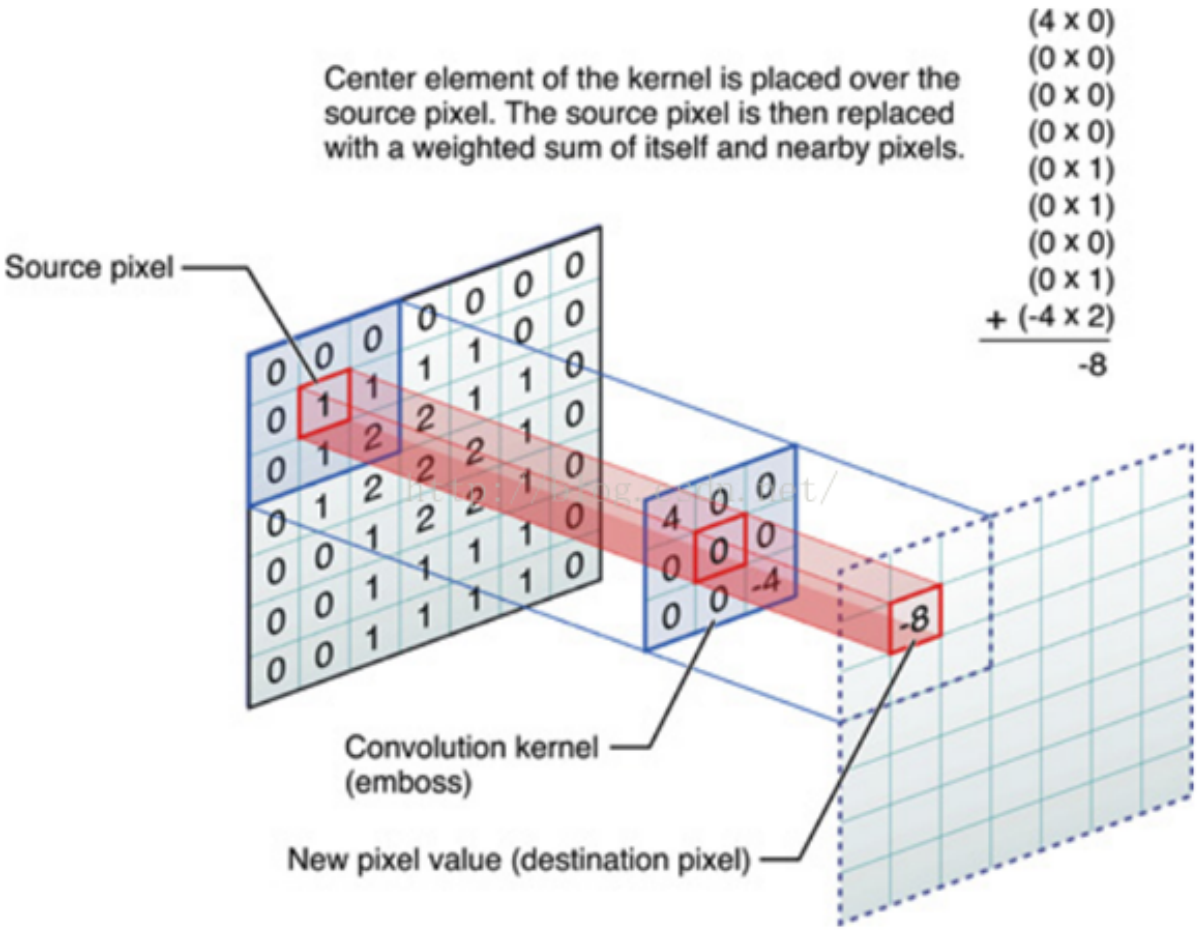
Convolution

2 layer input, 3 layer output convolution:



Convolution operation

2-D convolution

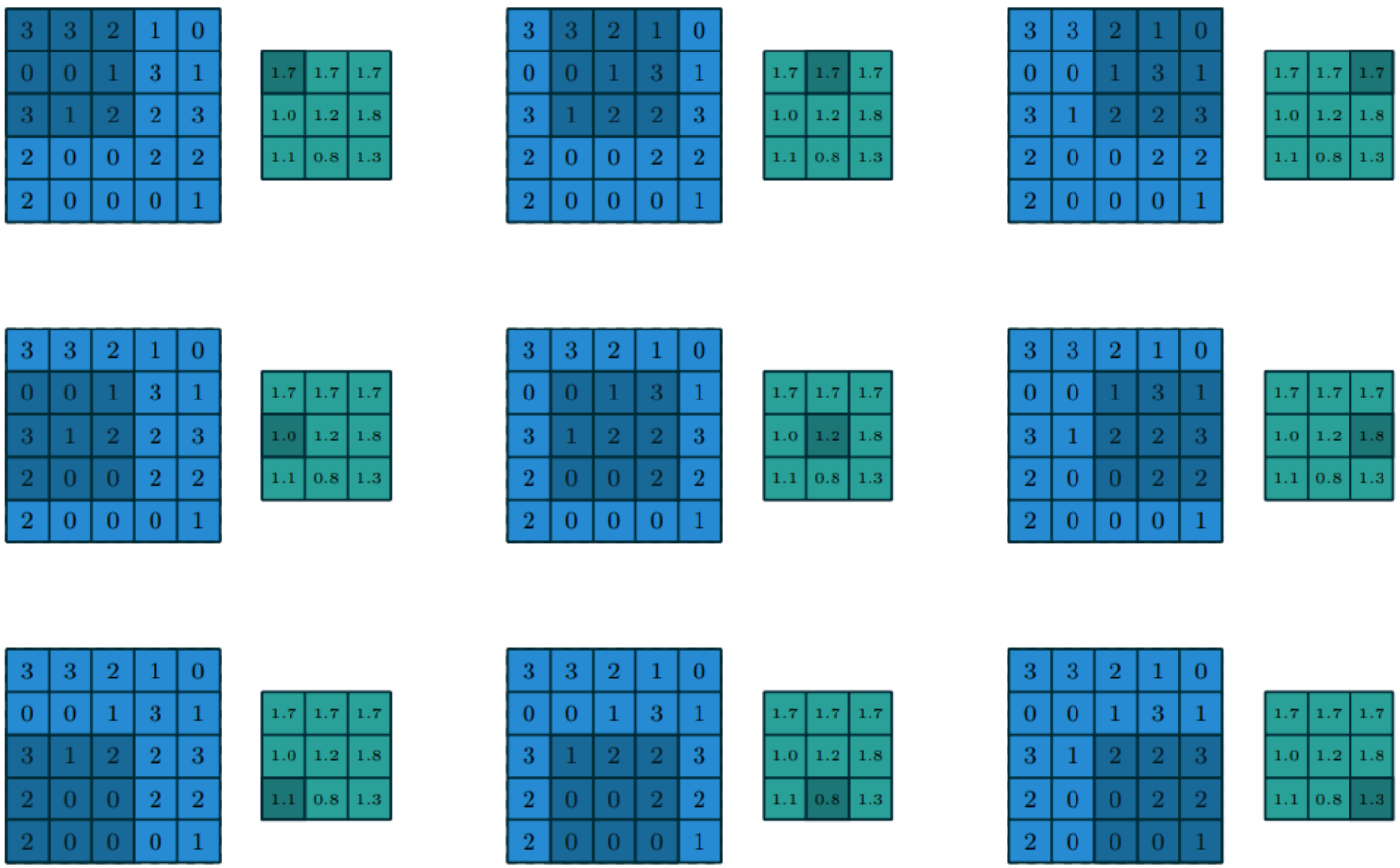


Comments

- Comments on convolution:
  - Locality: each neuron in the feature map takes output from small neighborhood of input layer neurons
  - Equivalence: the same transformation is applied by each neuron in the feature map
    - obtained by constraining sets of weights to each feature map layer neuron to be equal
  - This is feature extraction from a patch

Average pooling

Average 3x3 pooling:



Max pooling

Max 3x3 pooling:

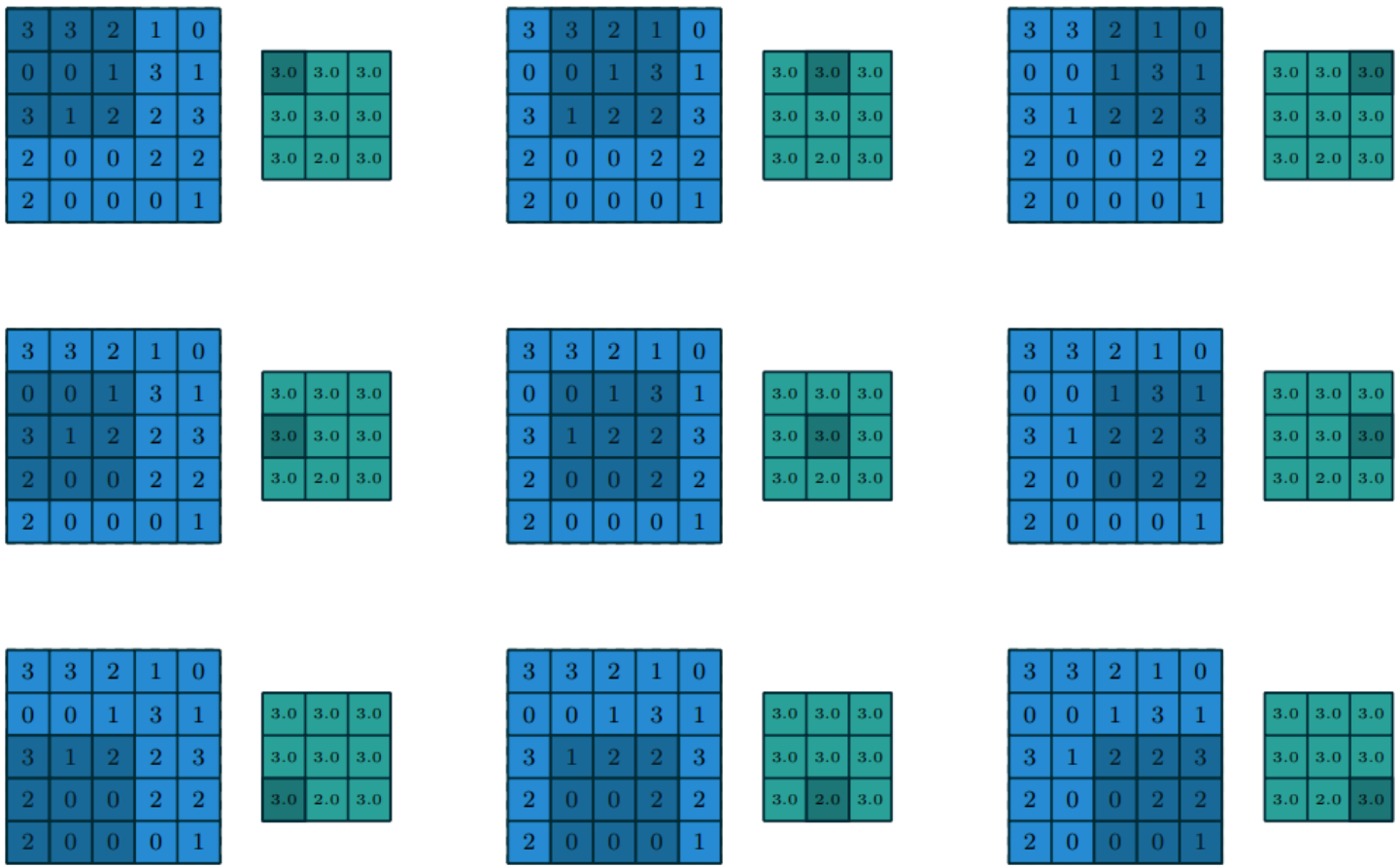
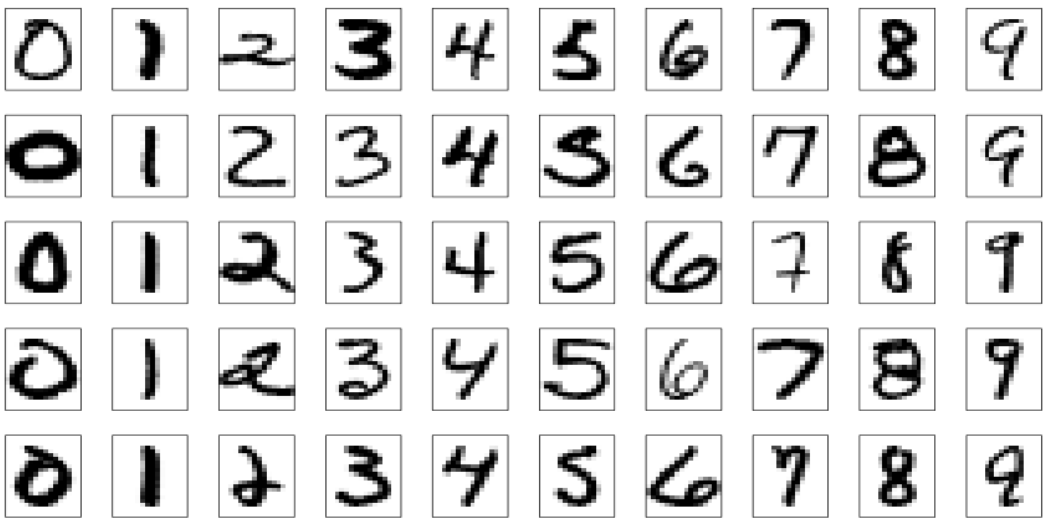


Table of Contents

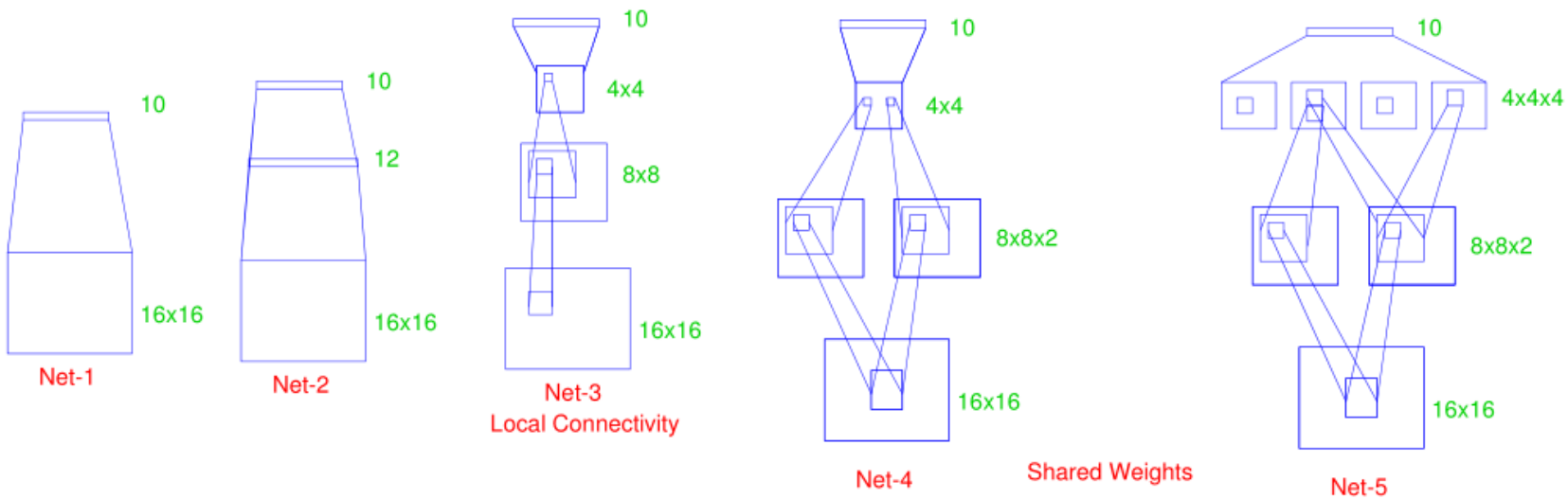
Case study (due to Hastie et al. The Elements of Statistical Learning)

ZIP code recognition task



Neural network structures

Net1: no hidden layer
Net2: 1 hidden layer, 12 hidden units fully connected
Net3: 2 hidden layers, locally connected
Net4: 2 hidden layers, locally connected with weight sharing
Net5: 2 hidden layers, locally connected, 2 levels of weight sharing



Results

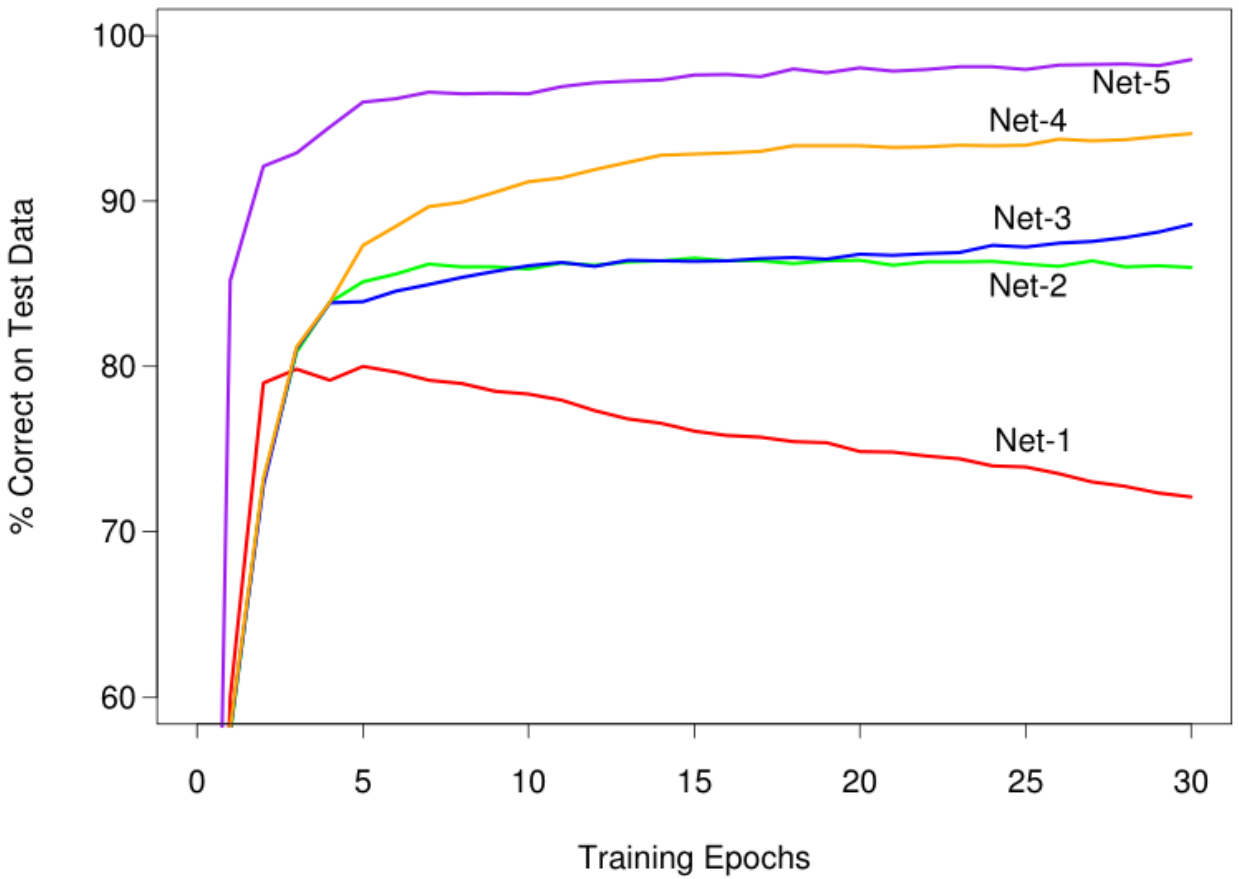
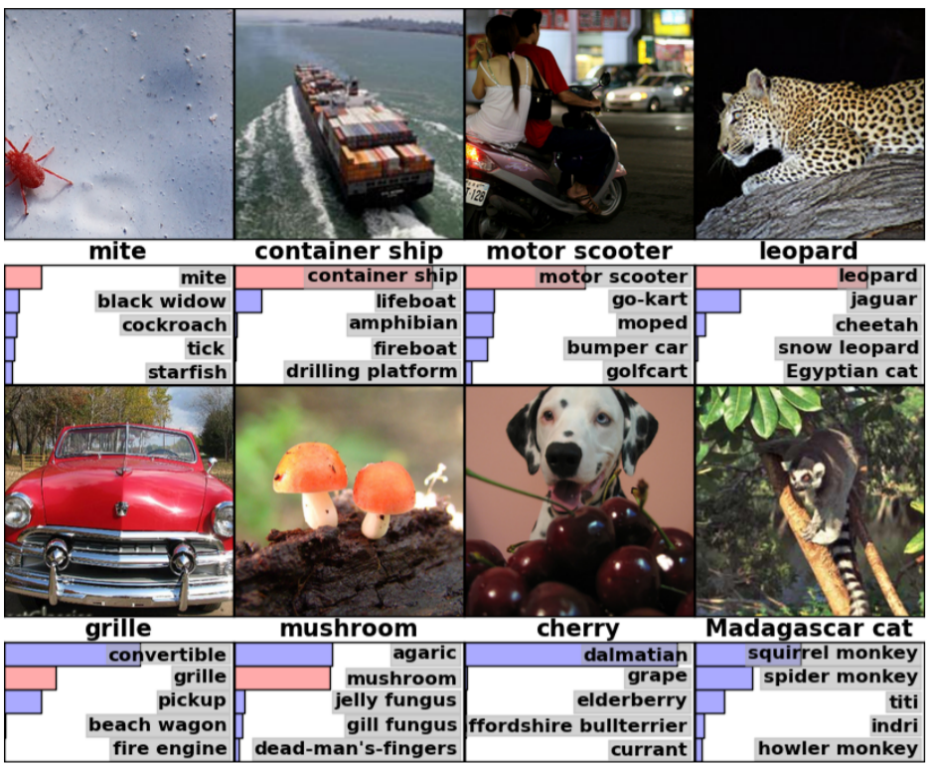

100
90
80
70
60
% Correct on Test Data
0
5
10
15
20
25
30
Training Epochs
Net-5
Net-4
Net-3
Net-2
Net-1

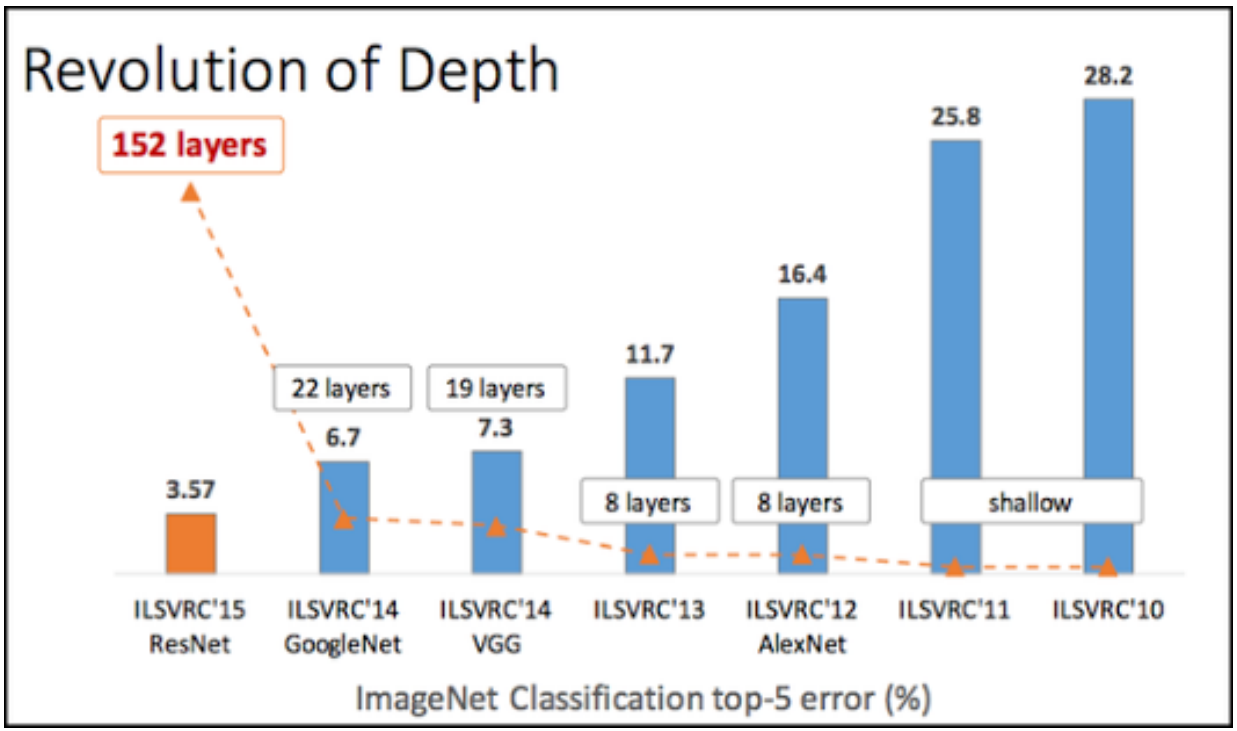
Table of Contents

ImageNet classification challenge

- 1000 unambiguous classes (including 120 dog breeds!).
- >1 million hand annotated images.
- Classifiers evaluated by top-5 accuracy
  - is the true class present among top-5 predictions?



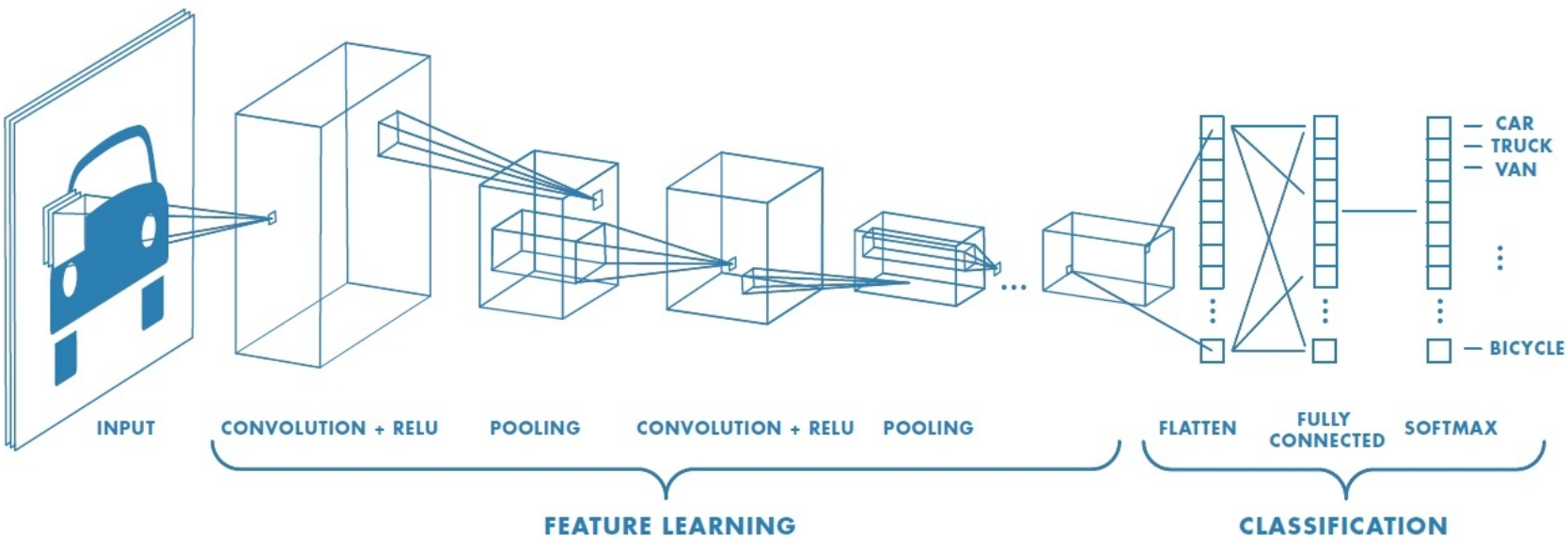
ImageNet challenge progress



- Starting from 2012 - triumph of deep convolutional networks.
- Human performance 3-15% (depending on acquaintance with the classes).[3]

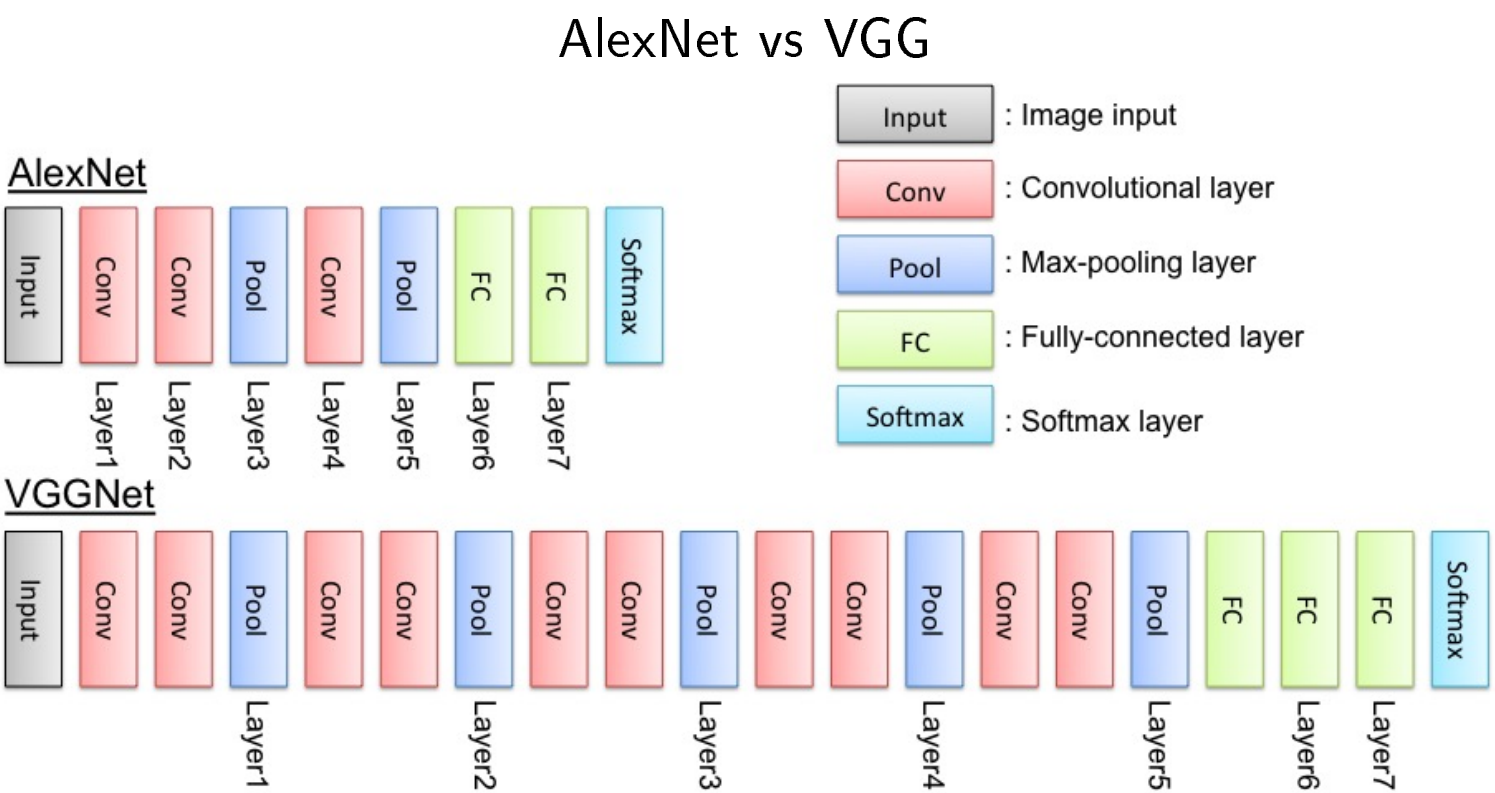
[3] Andrew Karpathy human test.

Convolution network



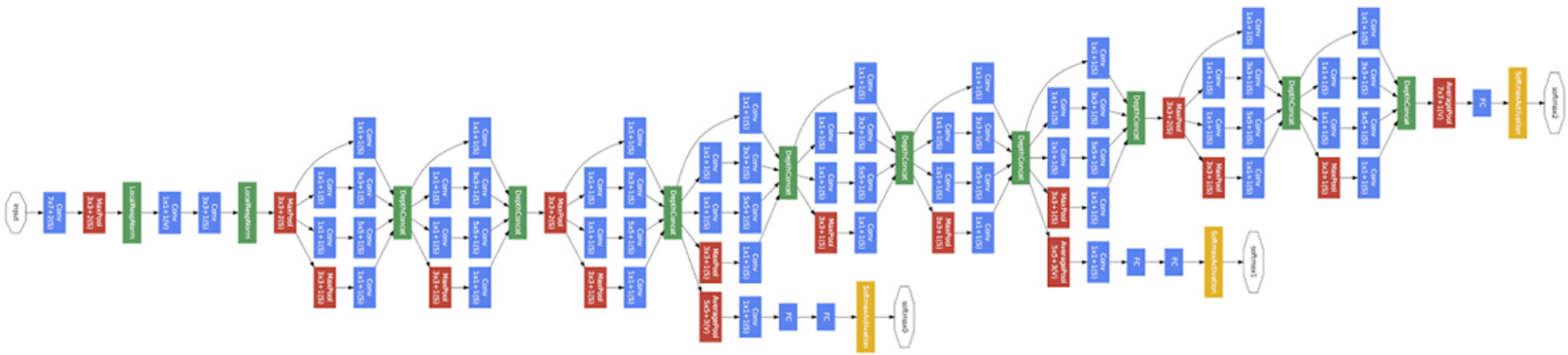
- Later layers learn more and more abstract features.
- Receptive field (in terms of original image) of neurons from deeper layers is wider.

Major CNN architectures



Each layer is followed by ReLu non-linearity.

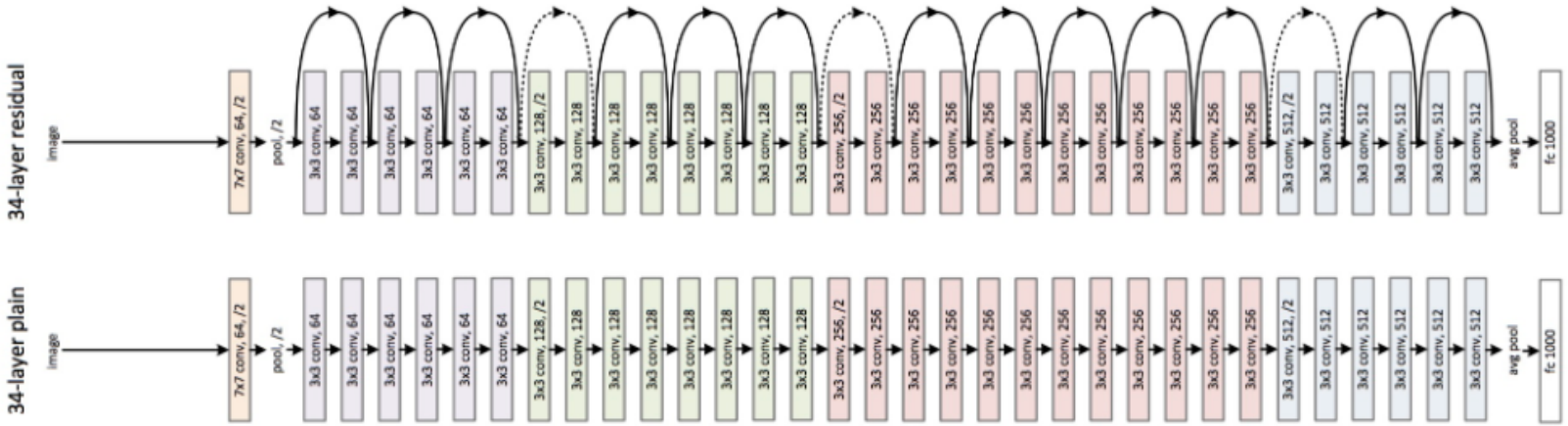
GoogleNet



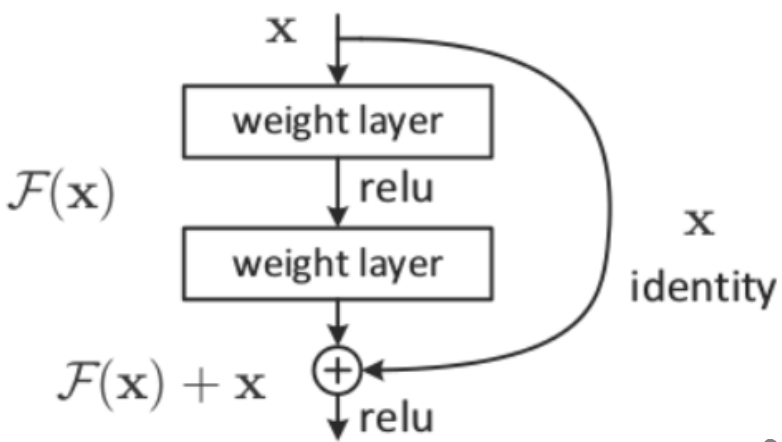
- add intermediate outputs during training
- reduce computation and # parameters by 1x1 convolutions

ResNet

ResNet vs. plain network



ResNet building block:



Skip identity connections allow:

- better propagate gradient backwards
- allow more natural initialization of weight layers
  - so that they are almost constant

VGG network[4]

Winners of ImageNet Challenge 2014!

- Data preprocessing - extract mean for R,G,B channels from all pixel intensities.
- Key idea: gradually reduce size and increase receptive field:
  - Filters with a very small receptive field: 3x3, 1x1.
  - Padding to keep original size (1 pixel for 3x3 conv)
  - Stride 1 for conv
  - Max-pooling is performed over a 2×2 pixel window, with stride 2.
- All hidden layers are followed by ReLU

[4] 2015 - Very deep convolutional networks for large-scale image recognition - Simonyan et al.

VGG details

- Optimization: SGD with momentum
  - learning rate decreased 3 times.
- Parallelization over minibatches
  - gradients are then averaged
- First 2 fully connected layers:
  - weight decay regularization
  - dropout regularization
- Train more shallow net, then with learned weights initialize deeper network.
- Dataset augmentation: random scaling, cropping.

Table of Contents

Definition of data augmentation

Data augmentation - extend training set $\{(x_n, y_n)\}_{n=1}^{N}$ with $\{(f_\theta(x_m), y_m)\}_{m=1}^{M}$ where

- $f_\theta(x)$ is label preserving transformation
  - in case of image: horizontal flipping, cropping, scaling, small rotation, small change in brightness, contrast, saturation, hue.
  - it may combine several transformations.
- θ are randomly sampled parameters of the transformation
- M is amount of new samples.

Applications of data augmentation

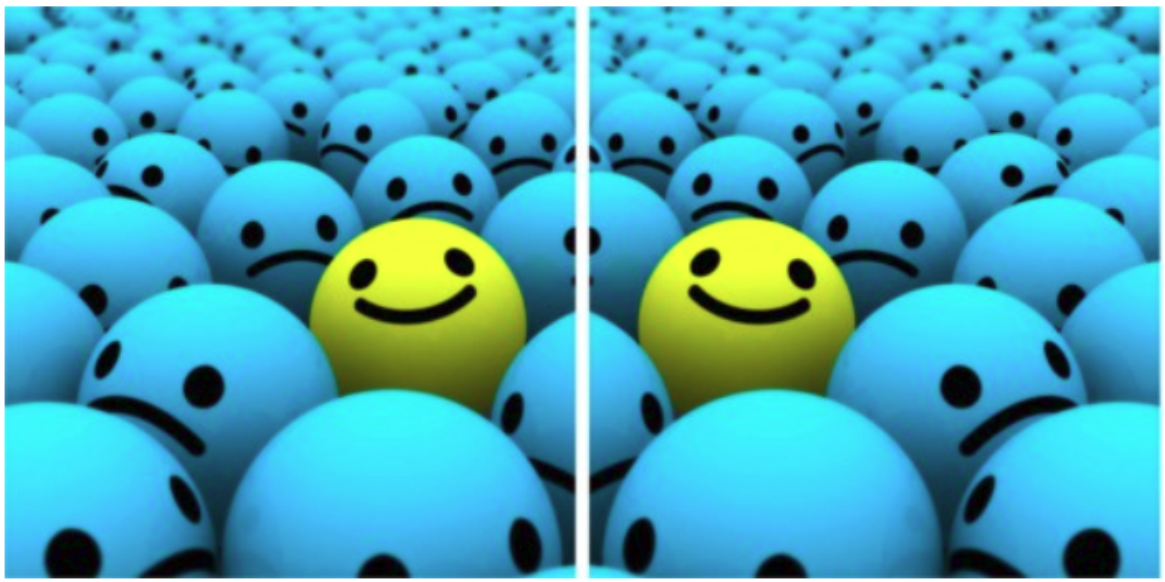
- Enlarge training set - estimate network parameters better.
  - feasible to estimate many parameters
  - less overfitting
- Enforce prediction invariance to transformation.
  - transfer learning: $f_\theta(\cdot)$ maps training domain to target domain.
    - e.g. have labelled *day-time* images, but need *night-time* classification.

#objects vs. #parameters[5]

	VGGNet	DeepVideo	GNMT
Used For	Identifying Image Category	Identifying Video Category	Translation
Input	Image 	Video 	English Text 
Output	1000 Categories	47 Categories	French Text
Parameters	140M	~100M	380M
Data Size	1.2M Images with assigned Category	1.1M Videos with assigned Category	6M Sentence Pairs, 340M Words
Dataset	ILSVRC-2012	Sports-1M	WMT'14

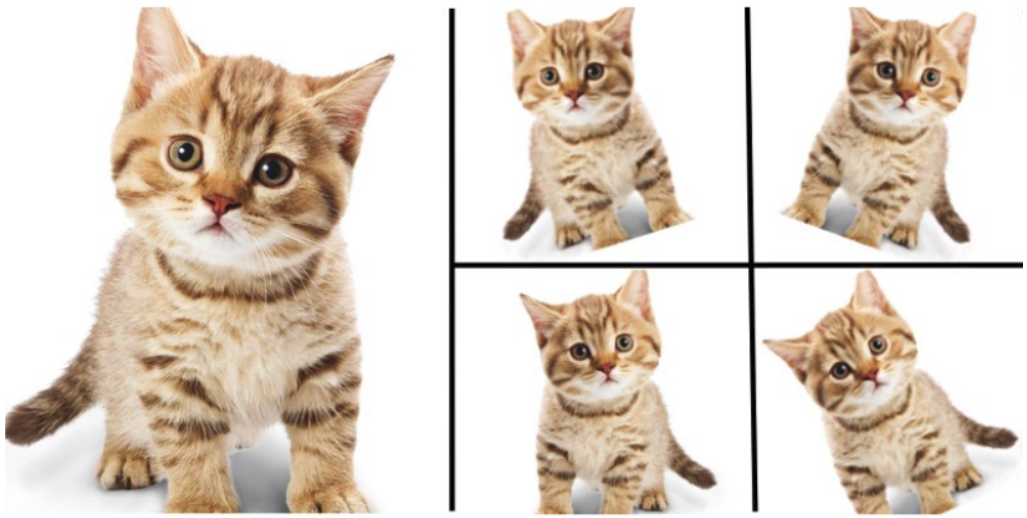
[5] Image source.

Horizontal flipping[6]



[6] Image source.

Rotations[7]

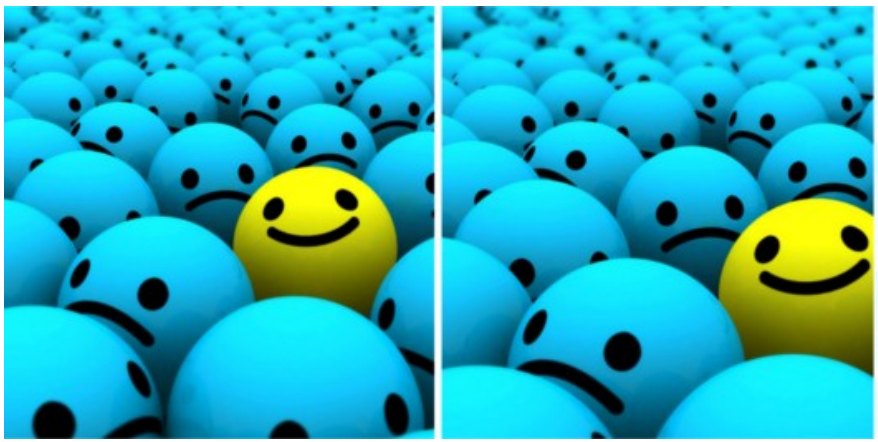


[7] Image source.

Translations[8]

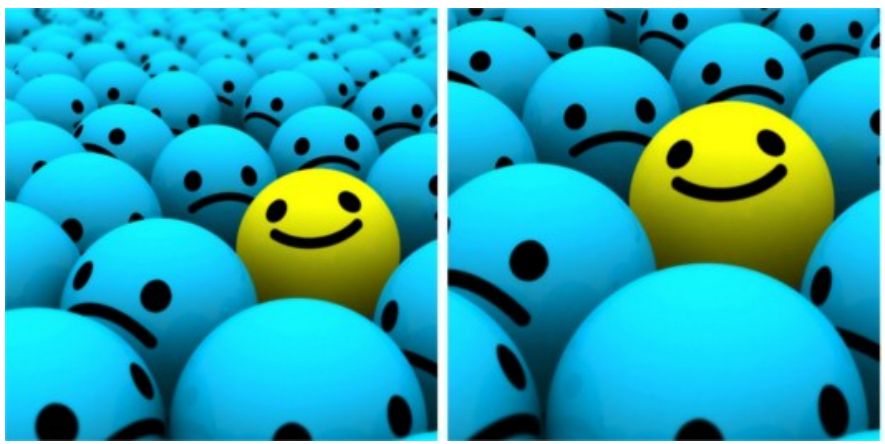
[8] Image source.

Scaling[9]



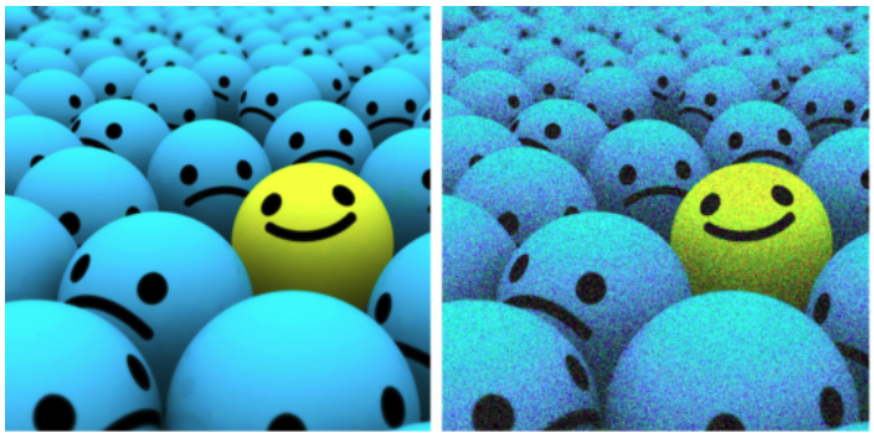
[9] Image source.

Cropping[10]



[10] Image source.

Adding noise (Gaussian, salt&pepper, etc.)[11]



[11] Image source.

Other augmentation methods

- Randomly adjust:
  - brightness (average level)
  - contrast (std. deviation of levels)
  - hue (add random offsets to RGB channels)
- Fill random patches with constant or noise
  - like dropout on input

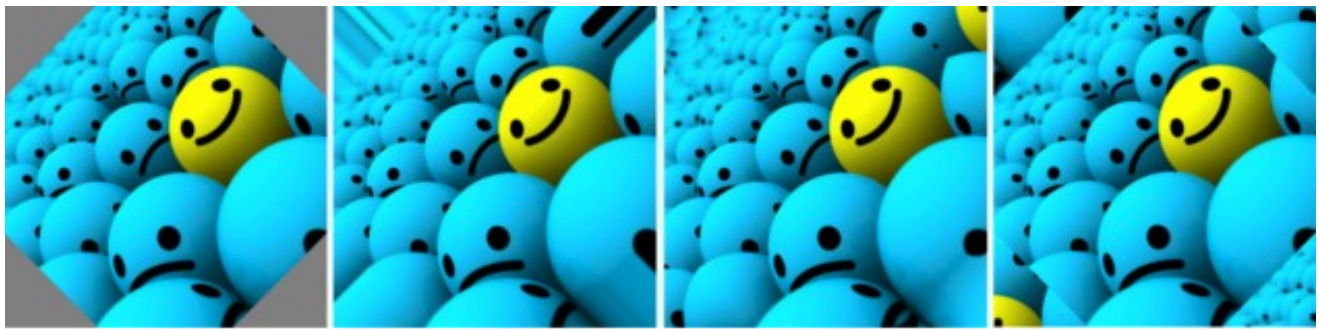
4 Data augmentation methods

- Padding unknown data

Transformation uncovers unknown data[12]

[12] Image source.

Methods to pad unknown data[13]



Unknown data padding methods (left to right):

- constant
- extend edge values
- mirror
- wrap around

[13] Image source.

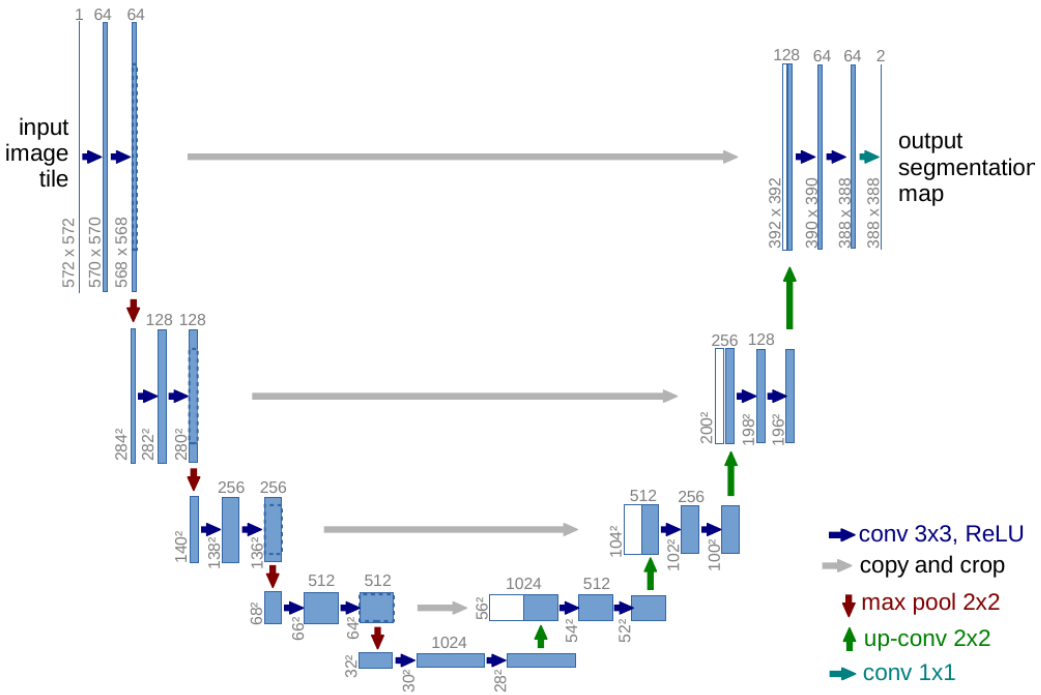
Table of Contents

Image segmentation[14]

- Segmentation - classification of every pixel of the image.
- Applications:
  - surveillance systems, autonomous driving, image classification, activity recognition on videos, etc.
- Model needs:
  - high level features to reconstruct object type
  - low level features to reconstruct boundaries

[14] Picture source.

U-net architecture[15]



Horizontal numbers = #[channels]; vertical numbers = spatial size.
White blocks - copied output of earlier layers; up-conv - rescaling & convolution.

[15] Ronneberger et al [2015].

Discussion

Key ideas of U-net:

- preserve spatial info at each layer
  - use only convolution, pooling, scaling.
  - don't use vectorization & fully connected layers
- 1st half - encoder; 2nd half - decoder.
- Encoder aggregates wider and wider local information
  - creating more abstract features
- Decoder reconstructs local information from
  - more abstract features (green input on figure)
  - lower level features (gray input on figure)