Imperial College London

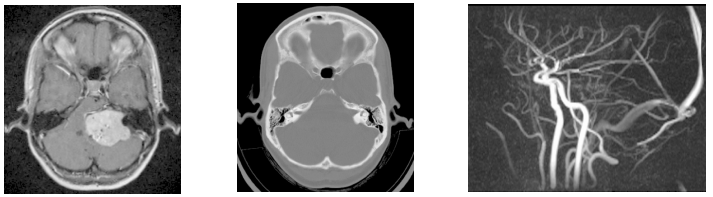
AI in Medicine I

Image Classification and/or Regression

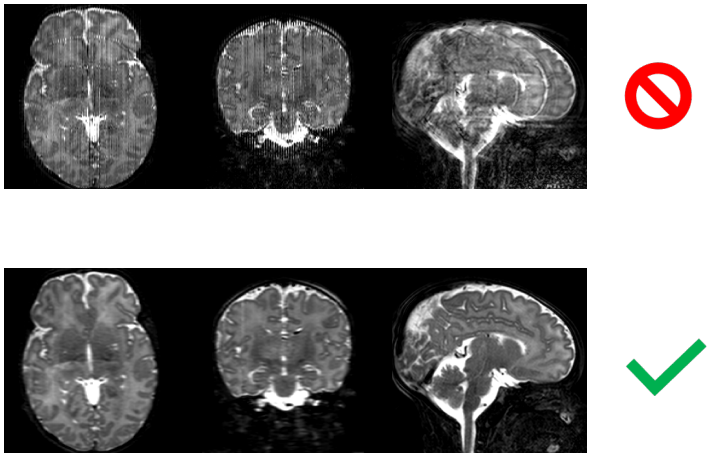
Daniel Rueckert
I31 – Chair for AI in Medicine and Healthcare
Faculty of Informatics and Medicine

Applications in medical imaging

- Image classification
  - Prediction of categorical parameters, e.g.
    - image modality (MR, CT, PET, etc)
    - image sequence (T1, T2, PD, etc)
    - quality of images (pass/fail)
    - disease state



MR CT Angio

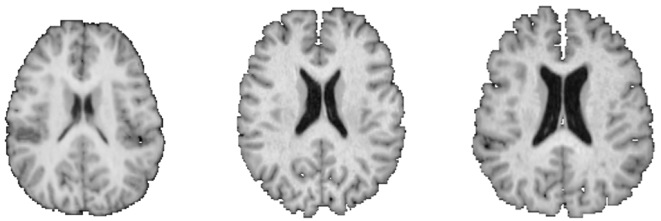


AD

Normal

Applications in medical imaging

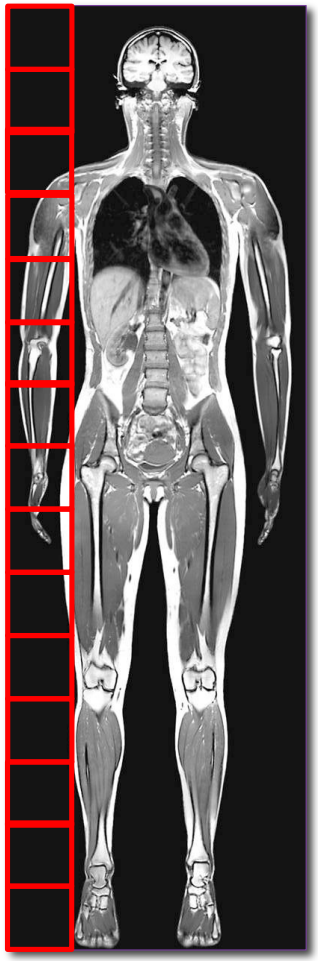
- Image classification
  - Prediction of categorical parameters, e.g.
    - image modality (MR, CT, PET, etc)
    - image sequence (T1, T2, PD, etc)
    - quality of images (pass/fail)
    - disease state
- Image regression
  - Prediction of quantitative parameters, e.g.
    - chronological age of the patient
    - quality of images (score between 0 and 1)



age = 7 age = 41 age = 60

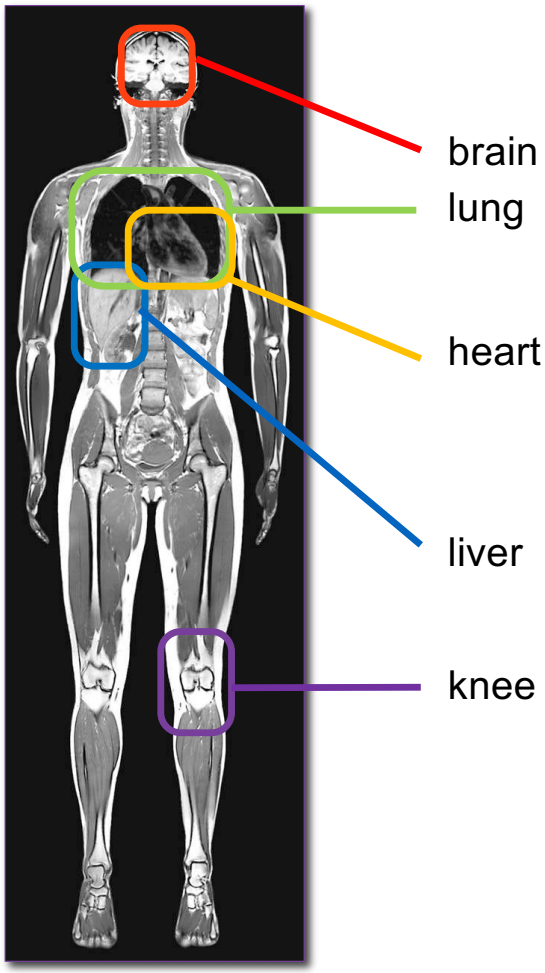
Applications in medical imaging

- Image classification
  - Prediction of categorical parameters, e.g.
    - image modality (MR, CT, PET, etc)
    - image sequence (T1, T2, PD, etc)
    - quality of images (pass/fail)
    - disease state
- Image regression
  - Prediction of quantitative parameters, e.g.
    - chronological age of the patient
    - quality of images (score between 0 and 1)
- Region classification
  - Anomaly detection
  - Anatomy or organ detection

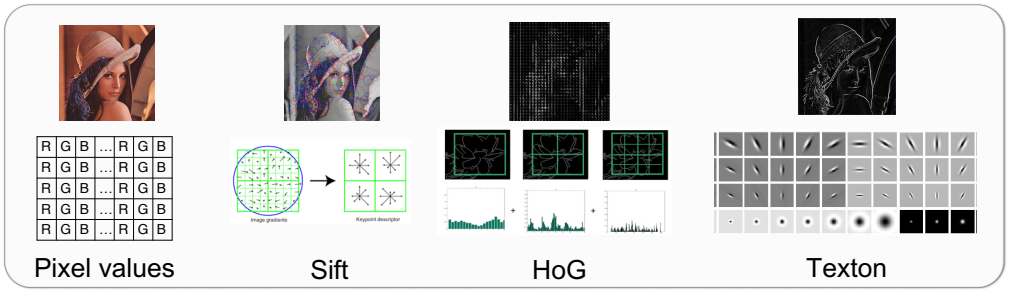


Applications in medical imaging

- Image classification
  - Prediction of categorical parameters, e.g.
    - image modality (MR, CT, PET, etc)
    - image sequence (T1, T2, PD, etc)
    - quality of images (pass/fail)
    - disease state
- Image regression
  - Prediction of quantitative parameters, e.g.
    - chronological age of the patient
    - quality of images (score between 0 and 1)
- Region classification
  - Anomaly detection
  - Anatomy or organ detection



Classical machine learning pipeline in imaging

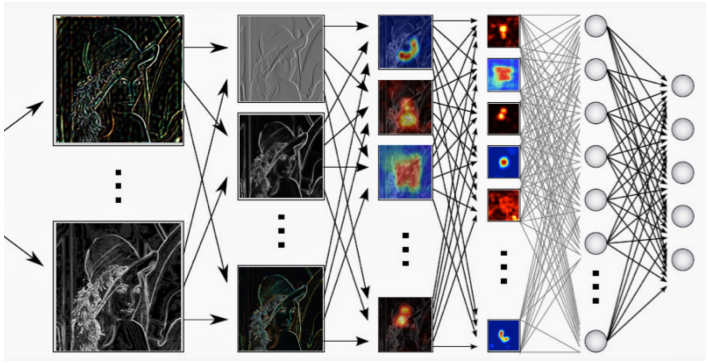


Feature extraction

ML model training/testing

Deep learning pipeline in imaging

Deep Neural Network



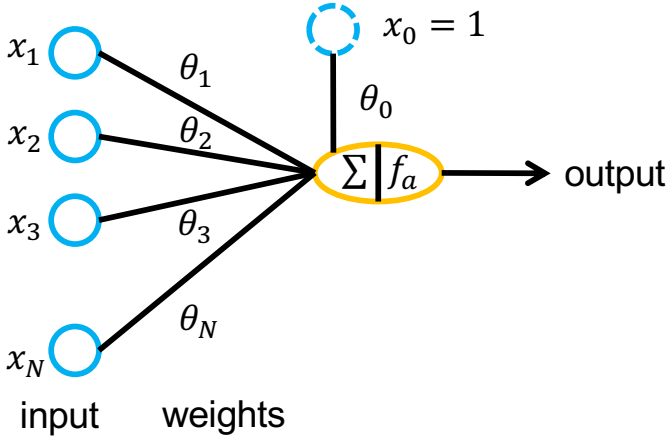
Images

$h_\Theta(\mathbf{I})$

Classification or prediction

Recap: (Artificial) Neural networks

Neuron activation: $\hat{y}(\boldsymbol{x}) = f_a\left(\sum_i x_i\theta_i + \theta_0\right)$

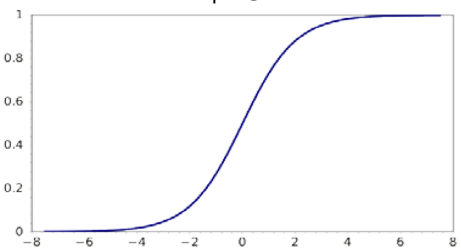


Loss: $\mathcal{L} = \frac{1}{N}\sum_i (y_i - \hat{y}_i)^2$

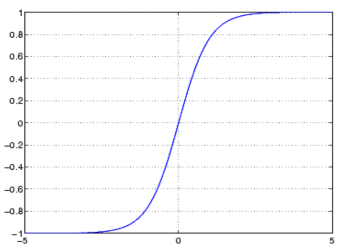
Learning: $\theta = \min_\theta \mathcal{L}(\theta, \mathbf{X}, \mathbf{Y})$ via gradient descent: $\theta_i^{t+1} = \theta_i^t + \frac{\partial \mathcal{L}}{\partial \theta_i^t}$

Activation functions:

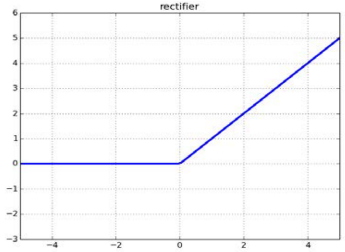
$S(x) = \frac{1}{1+e^{-x}}$



$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$

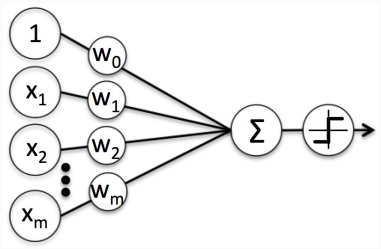


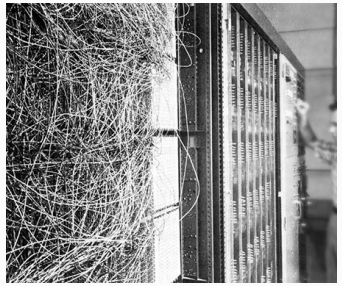
$f(x) = \max(0, x)$



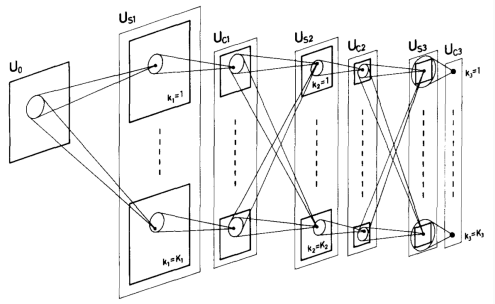
Some history on neural networks…

- **Perceptron** [Rosenblatt 1958]:





- **Neocognitron** [Fukushima 1980]:



- **Back-propagation** [Rumelhart 1985]:

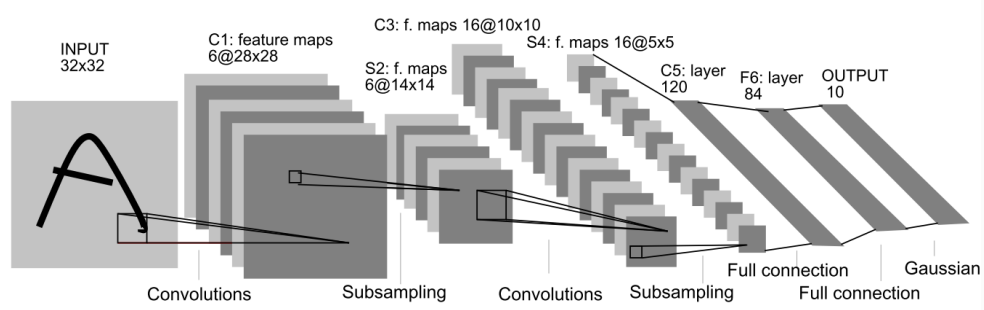
$$\Delta_p w_{ji} = \eta \delta_{pj} o_{pi}$$

$$\delta_{pj} = (t_{pj} - o_{pj}) f'_j(net_{pj})$$

$$\delta_{pj} = f'_j(net_{pj}) \sum_k \delta_{pk} w_{kj}$$

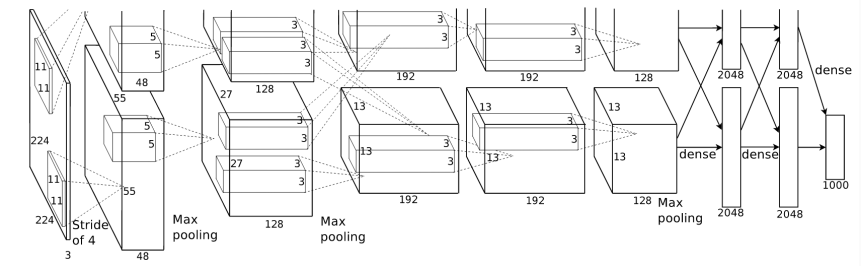
More recently: The deep learning (r)evolution

- **LeNet [LeCun1989, 1998]:**

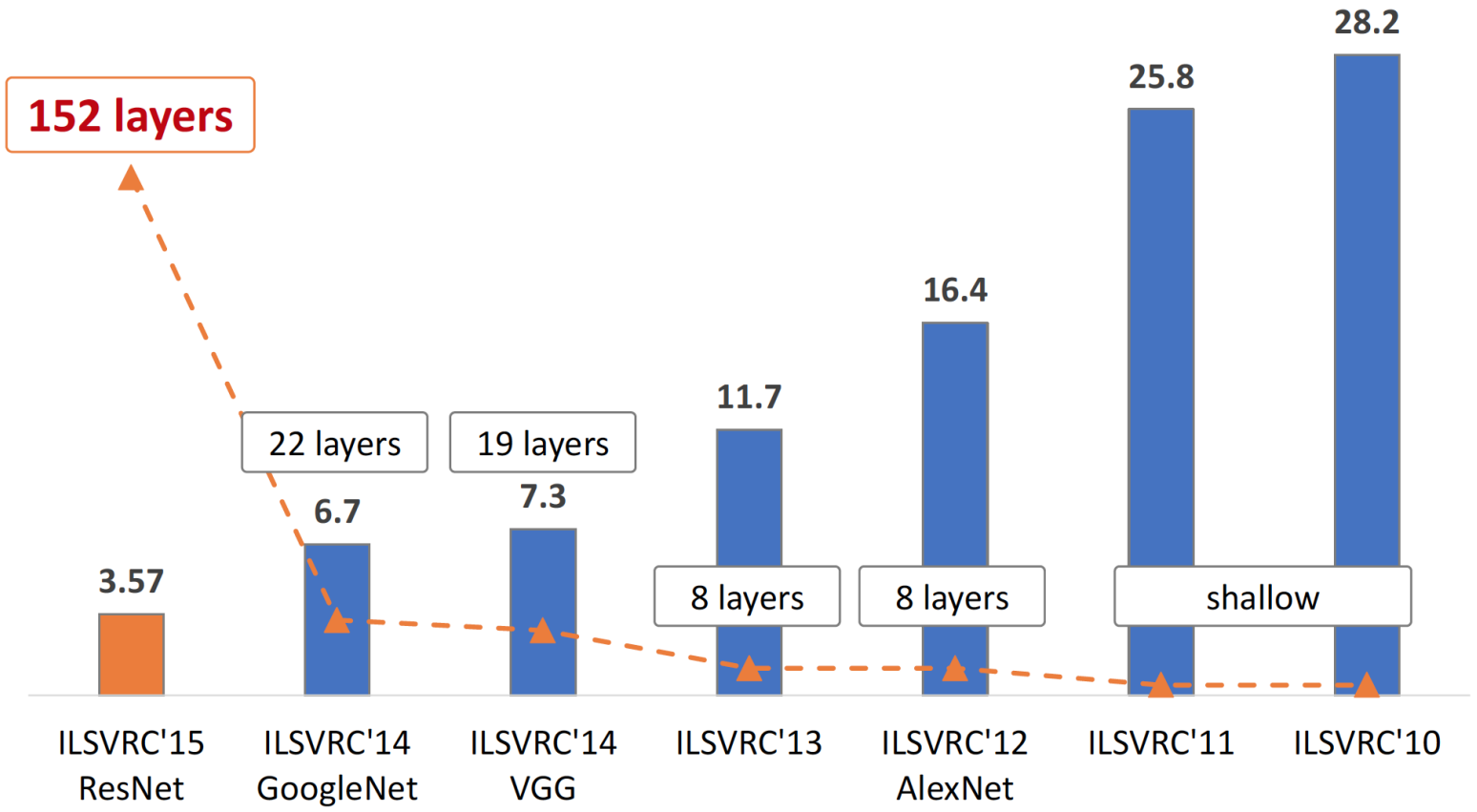


- **General Purpose GPU [CUDA 2007]:**

- **AlexNet [Krizhevsky 2012]:**

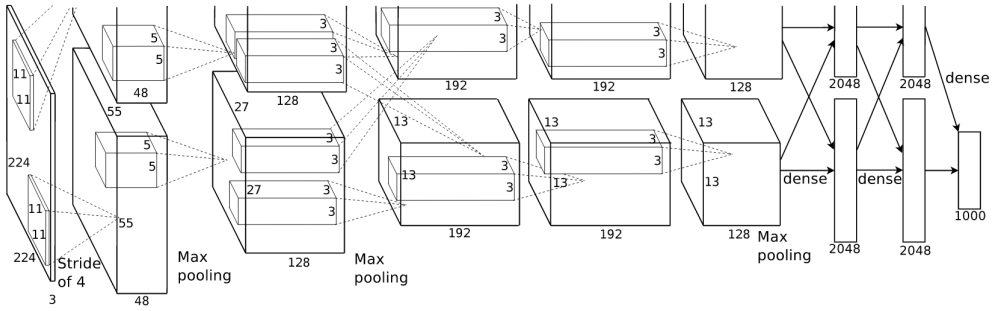


More recently: The deep learning (r)evolution

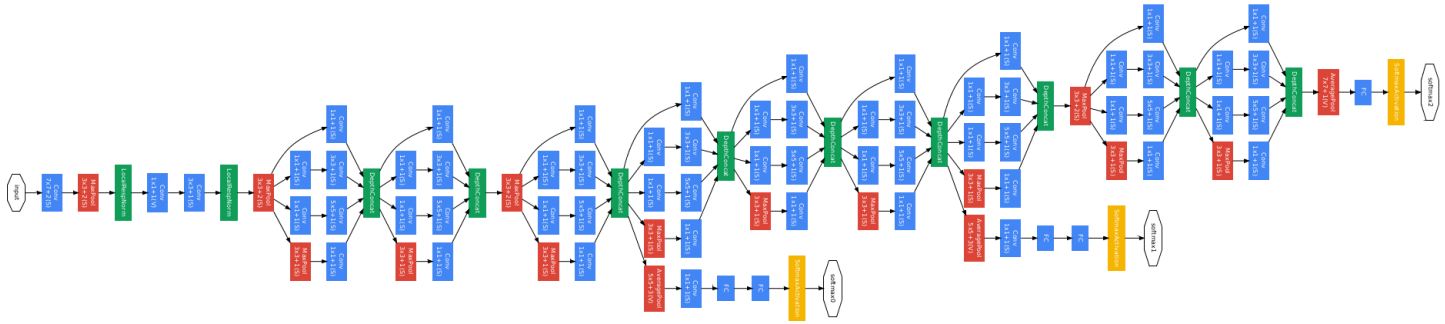


More recently: The deep learning (r)evolution

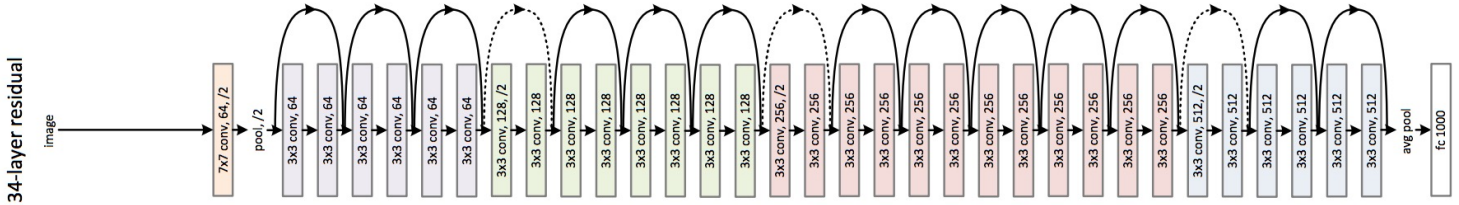
AlexNet



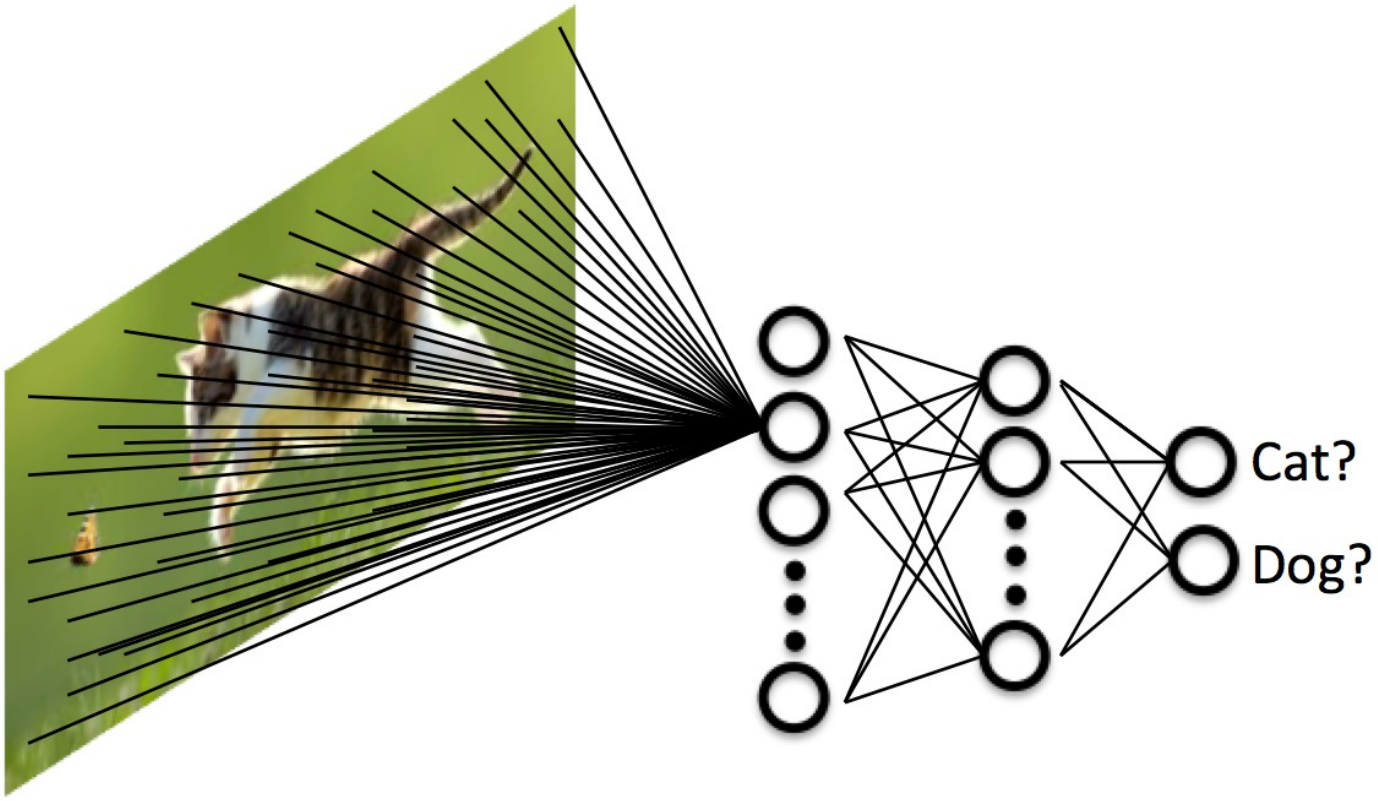
GoogLeNet



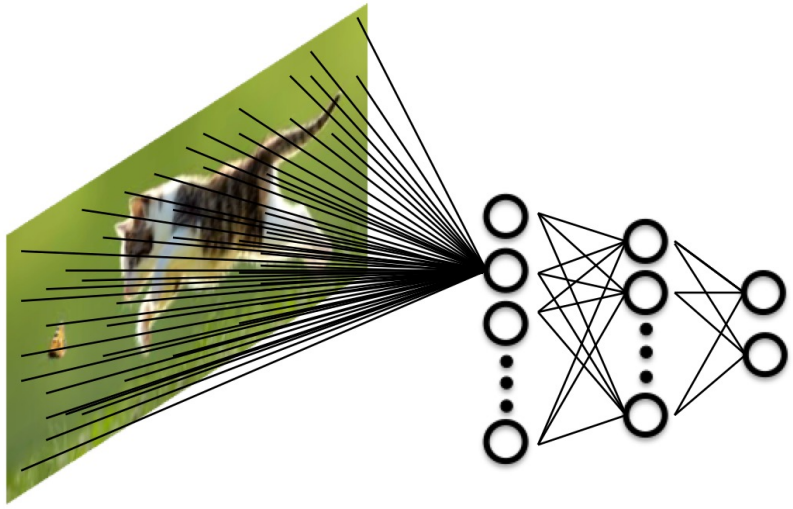
ResNet



Neural networks for image classification



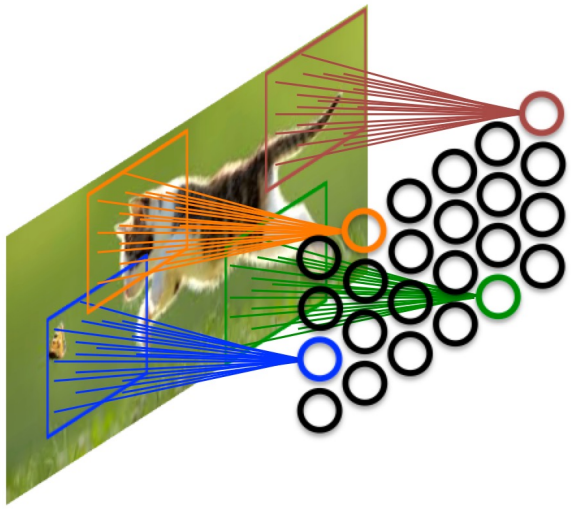
Neural networks: Fully connected networks



- Each neuron detects a different pattern.
- Too many weights for each neuron/pattern.
  - Example: 512 x 512 image = more than 260,000 weights/pattern.

Features are local: Only neighboring pixels are correlated!

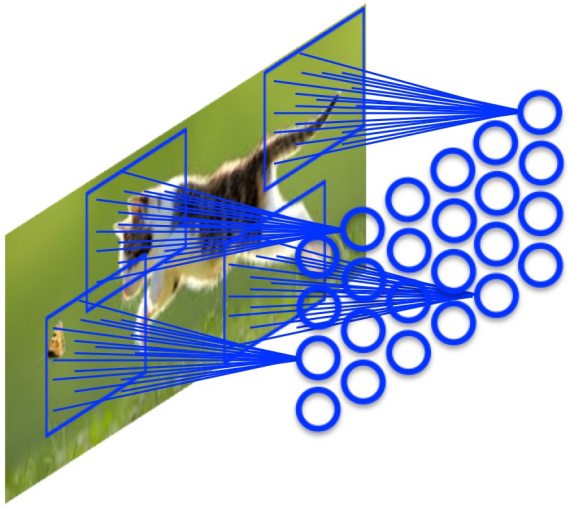
Neural networks: Locally connected networks



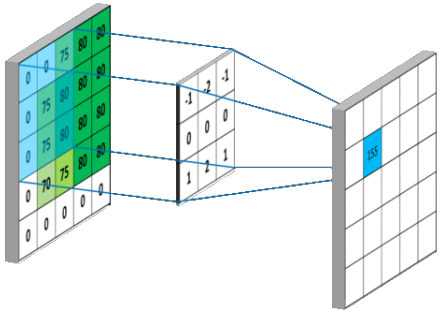
- Each neuron detects a different pattern.
- Kernel of a neuron: Spatially limited connections
  - Example: If kernel = 10x10, 100 weights/pattern.
  - Each neuron only detects a pattern at a certain location.

Stationarity in vision:
The same feature may appear anywhere in the image!

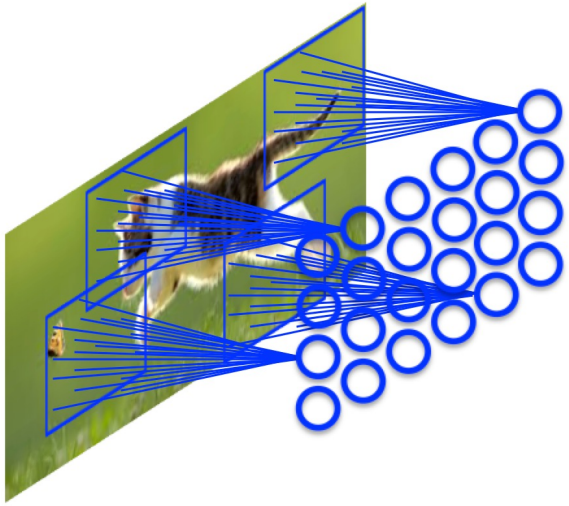
Neural networks: Locally connected networks with shared weights

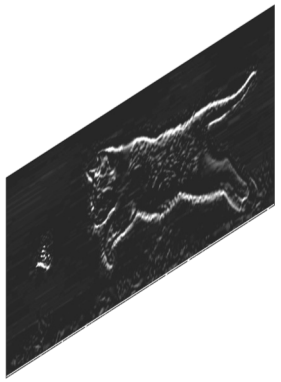


Convolution



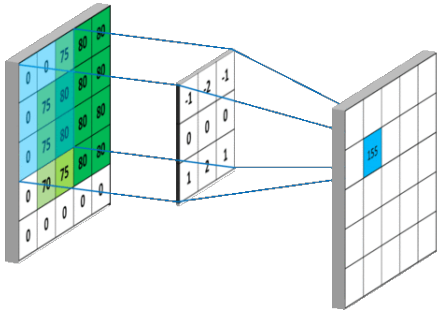
Locally Connected Networks with Weight Sharing



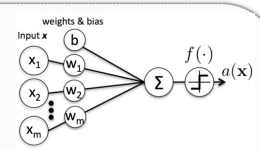


Activation function

Convolution



Single neuron:



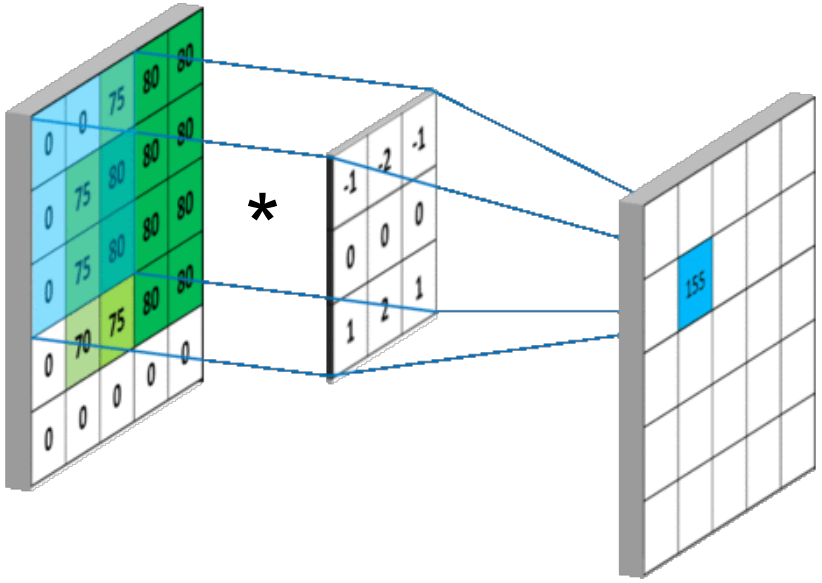
$$a(\mathbf{x}) = f_a\left(\sum_i \theta_i x_i + b\right)$$

Feature Map:

$$\mathbf{a}(\mathbf{x}) = f_a(\boldsymbol{\theta} * \mathbf{x} + b)$$

Recap: Convolution

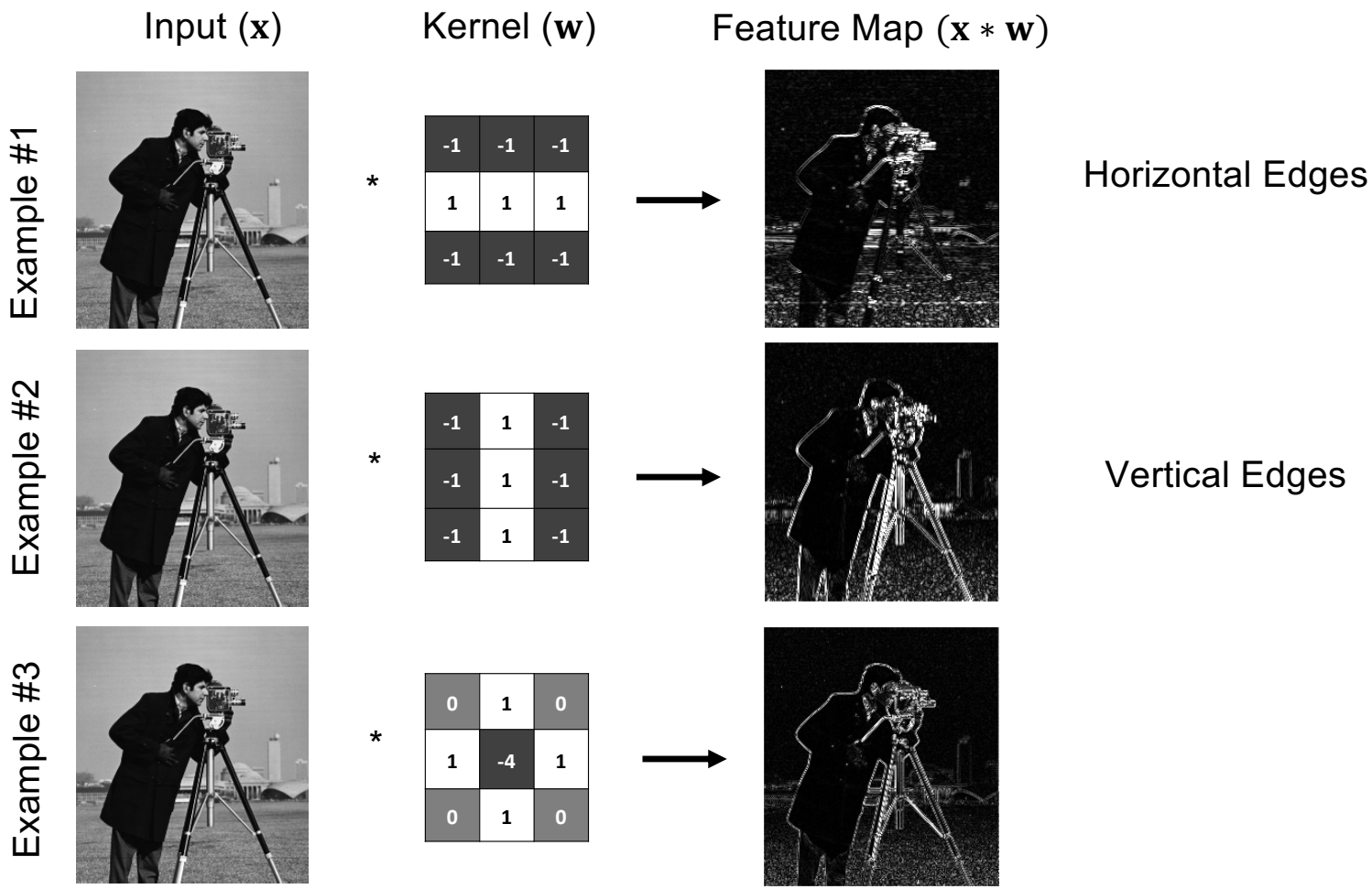
Input: $\boldsymbol{f}$ Kernel: $\boldsymbol{g}$ Output: $\mathbf{o} = (\boldsymbol{f} * \boldsymbol{h})$



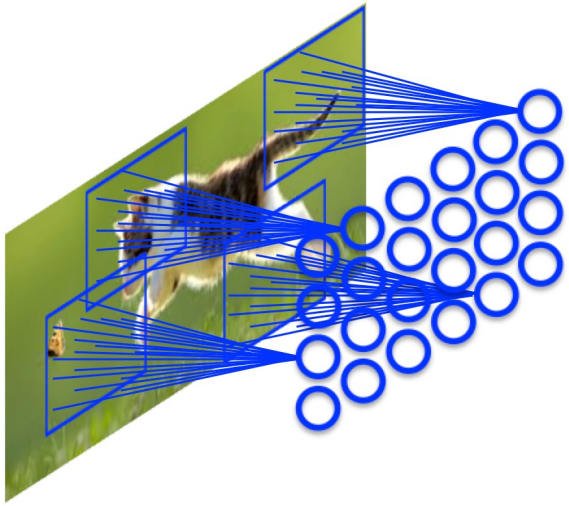
Assume 2D kernel $\boldsymbol{g}$ of dimensions $n \times n$:

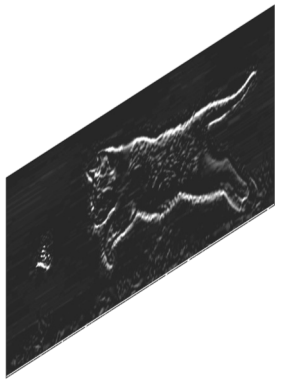
$$(\boldsymbol{f} * \boldsymbol{g})[x, y] = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \boldsymbol{f}\left(x - i + \left\lfloor \frac{n}{2} \right\rfloor, y - j + \left\lfloor \frac{n}{2} \right\rfloor\right) \boldsymbol{g}(i, j)$$

Recap: Convolution



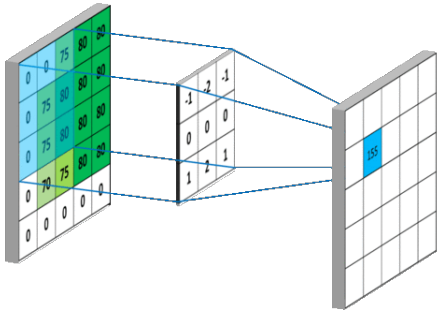
Locally Connected Networks with Weight Sharing



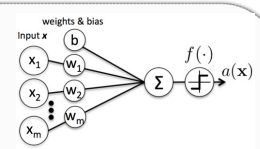


Activation function

Convolution



Single neuron:



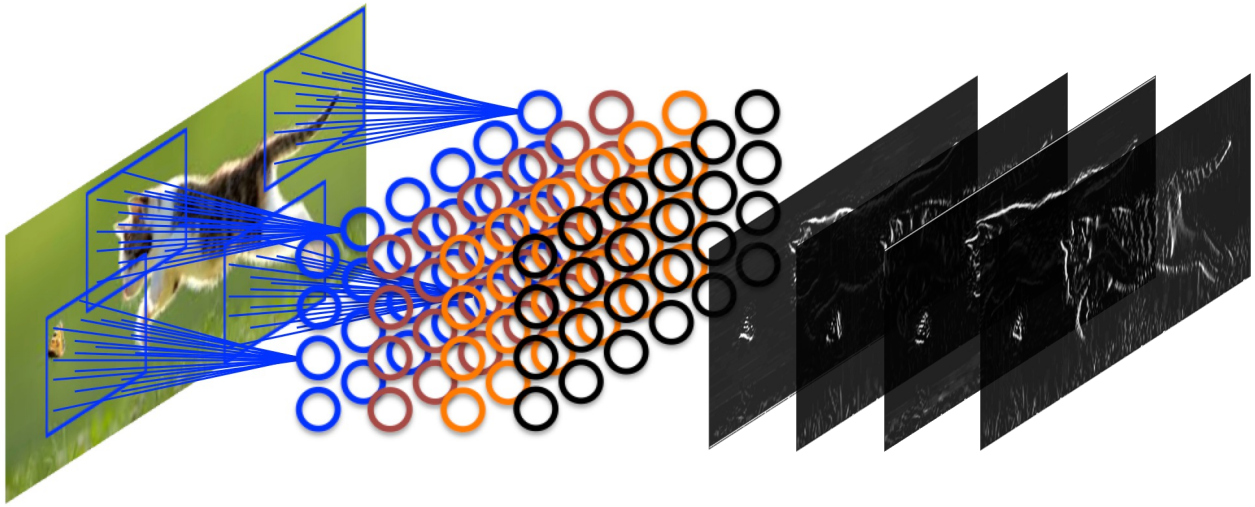
$$a(\mathbf{x}) = f_a\left(\sum_i \theta_i x_i + b\right)$$

Feature Map:

$$\mathbf{a}(\mathbf{x}) = f_a(\boldsymbol{\theta} * \mathbf{x} + b)$$

Feature map is just another image

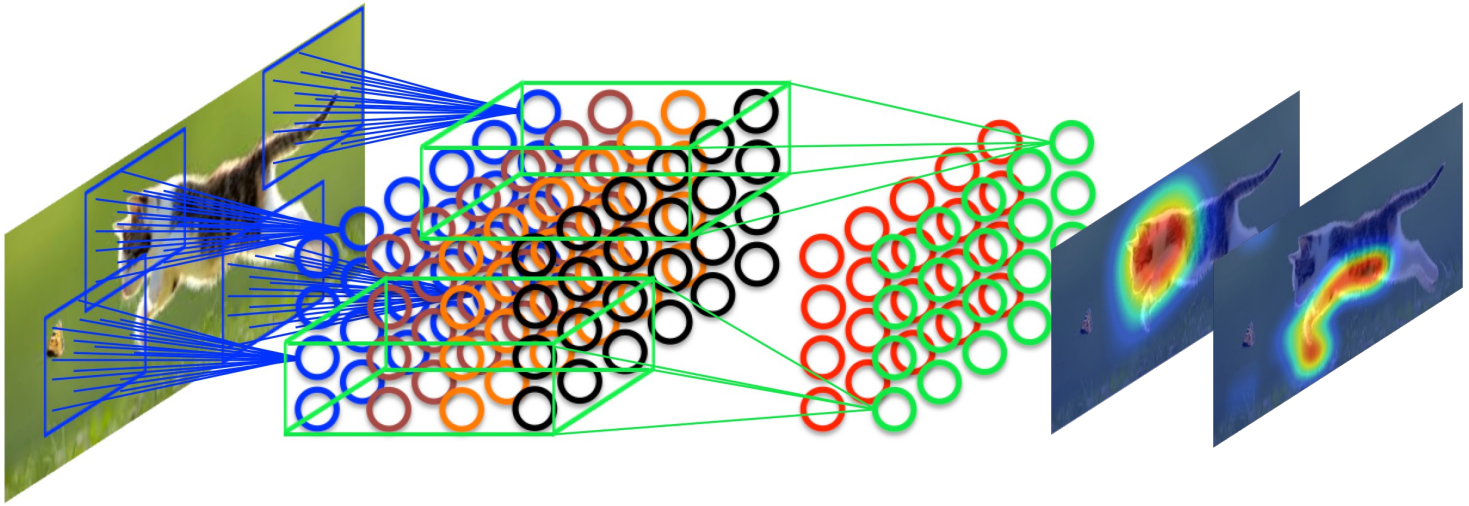
Convolutional neural networks



Convolutional Layer:

- Multiple Feature Maps – Detect multiple features per layer.
- A layer can be perceived as a multi-channel image (similar to RGB channels).

Convolutional neural networks



Convolutional Layer:

- Multiple Feature Maps – Detect multiple features per layer.
- A layer can be perceived as a multi-channel image (similar to RGB channels).
- Deeper layers process FMs (channels) of previous layer.
- Combine previous features to extract more complex representations.

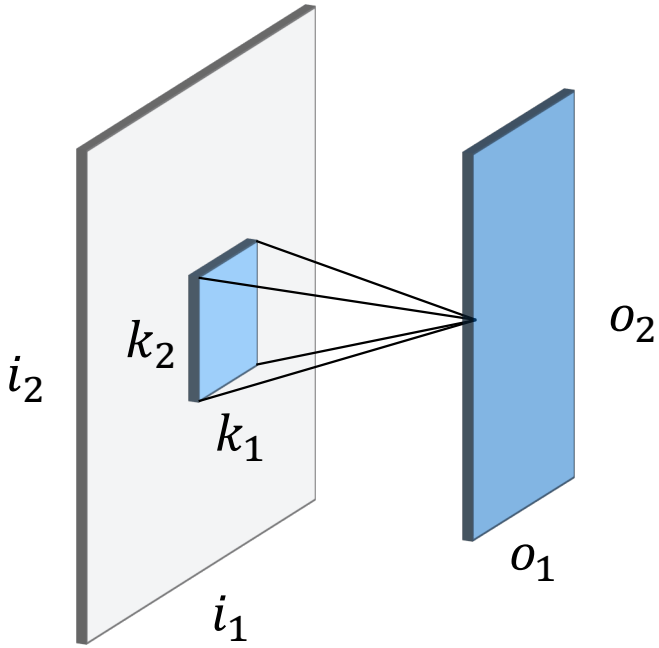
Convolutional layers

Q: What is the size of the feature map?

A: This depends on a number of parameters

- Input size along axis j: i_j
- Kernel size along axis j: k_j
- Zero padding (number of zeros concatenated at the beginning and at the end of an axis) along axis j: p_j
- Stride (distance between two consecutive positions of the kernel) along axis j: s_j

Convolutional layers



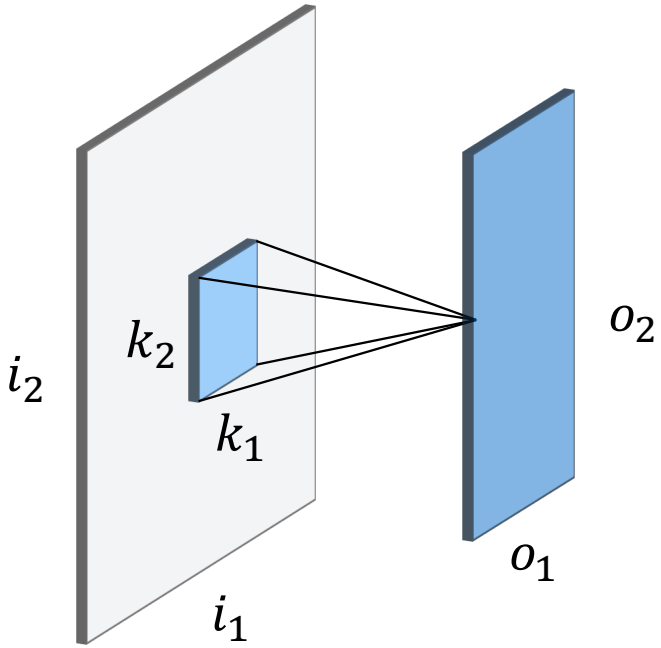
Assuming no padding, no stride

$o_j = i_j - k_j + 1$

Example: $i_j = 5, k_j = 3$

$o_j = 5 - 3 + 1 = 3$

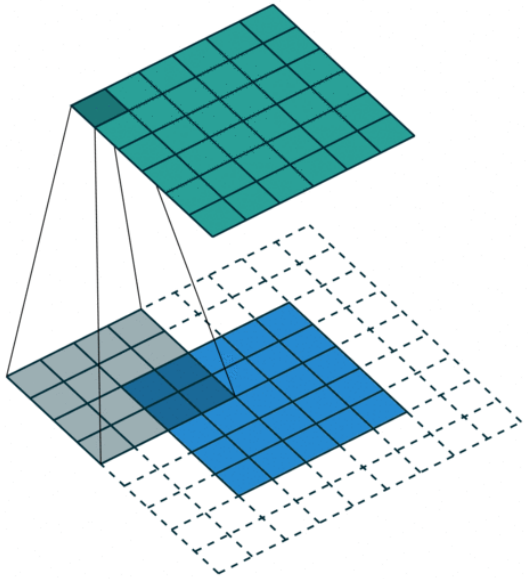
Convolutional layers



Assuming padding of size p, no stride

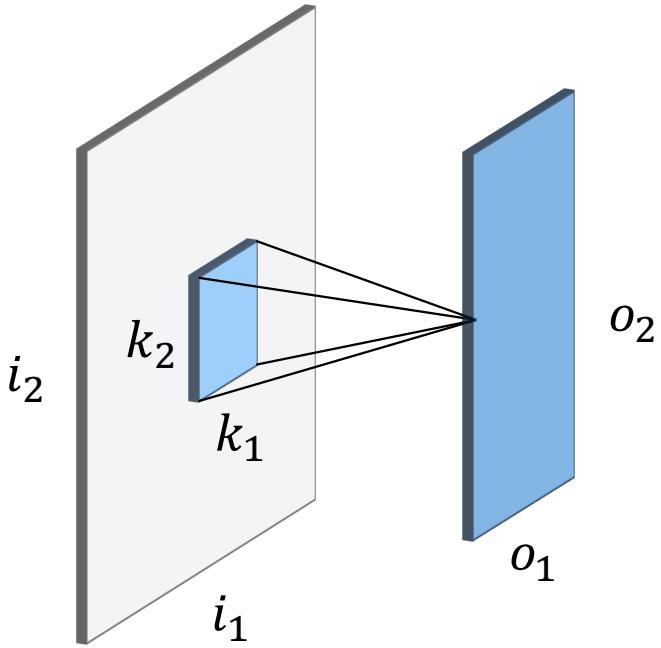
$$o_j = i_j - k_j + 2p + 1$$

Example: $i_j = 5, k_j = 4, p_j = 2$



$$o_j = 5 - 4 + 4 + 1 = 6$$

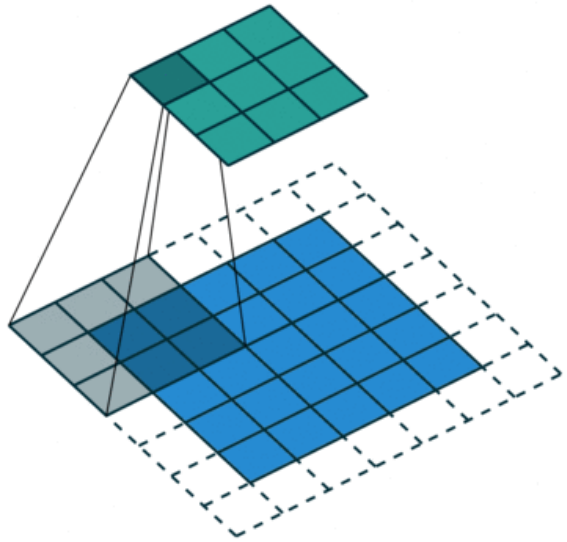
Convolutional layers



Assuming padding of size p, stride s_j

$$o_j = \lfloor (i_j - k_j + 2p_j)/s_j \rfloor + 1$$

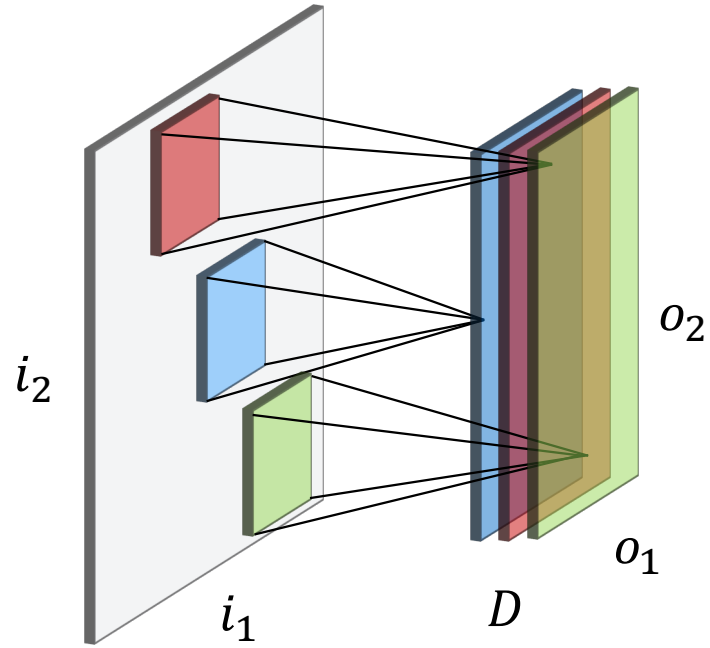
Example: $i_j = 5, k_j = 3, p_j = 1, s_j = 2$



$$o_j = \lfloor (5 - 3 + 2)/2 \rfloor + 1 = 3$$

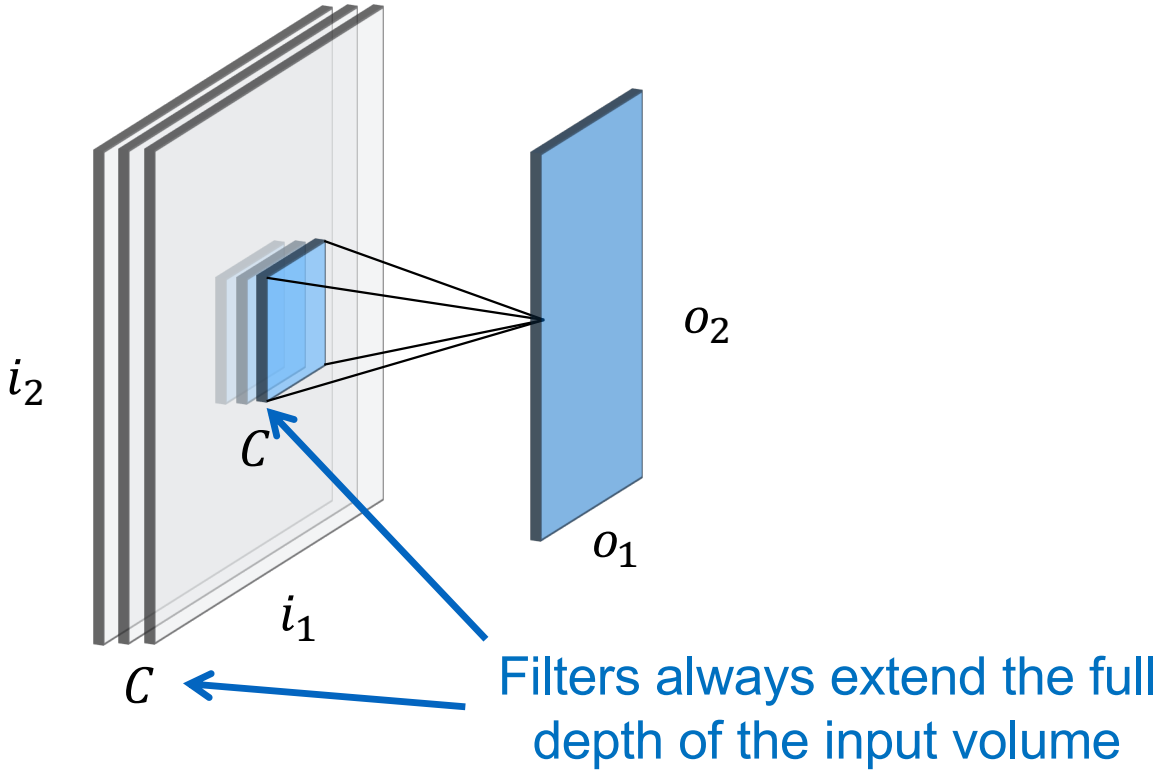
Convolutional layers

- As we have seen we can learn multiple kernels (or feature maps) at the same time by stacking D output feature maps



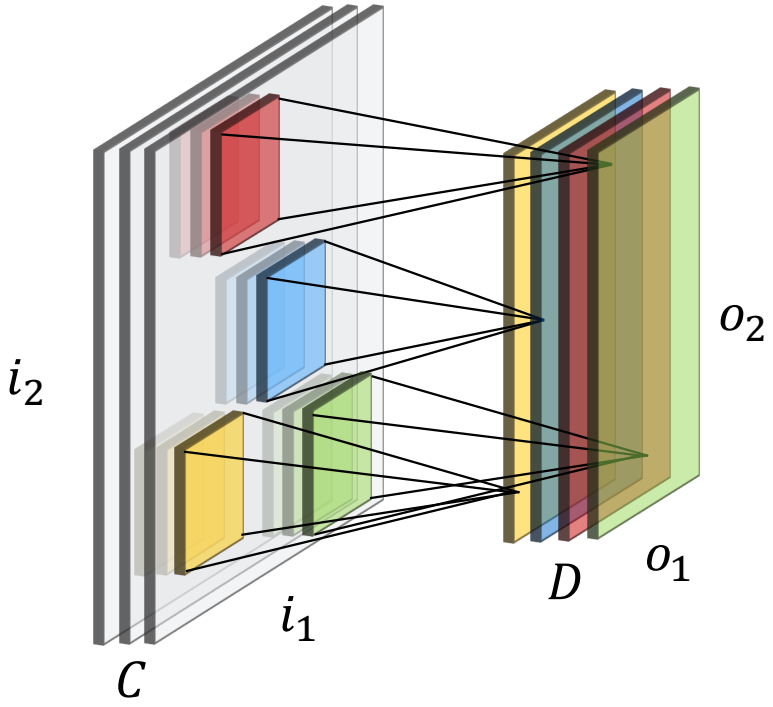
Convolutional layers

- However, the input to a convolutional layer may also have C input channels, e.g. for a RGB image $C = 3$.



Convolutional layers

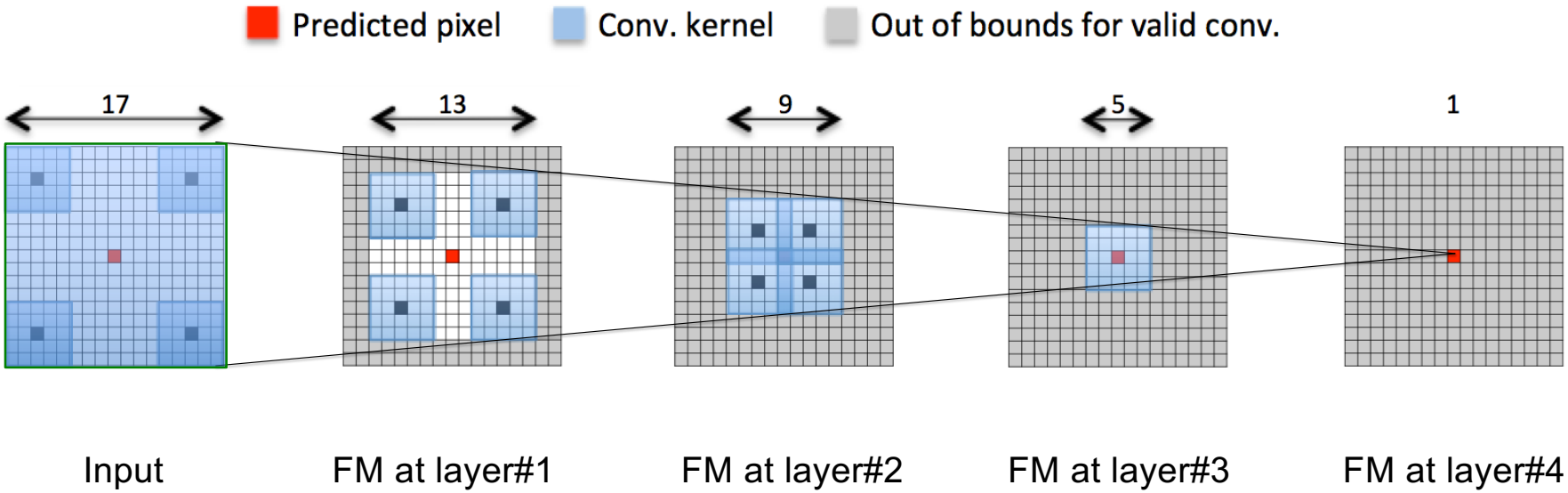
- Example: RGB image (input layer $C = 3$) with four filters (output layer $D = 4$)



What have we gained?

- Stack multiple filters to get a filter bank
- Layer with $D = 4$ filters with 5 x 5 x 3 kernel size:
  - 5 x 5 x 3 x 4 = 300 weights
- Convolution: Independent of image size
- Much more training data for weights

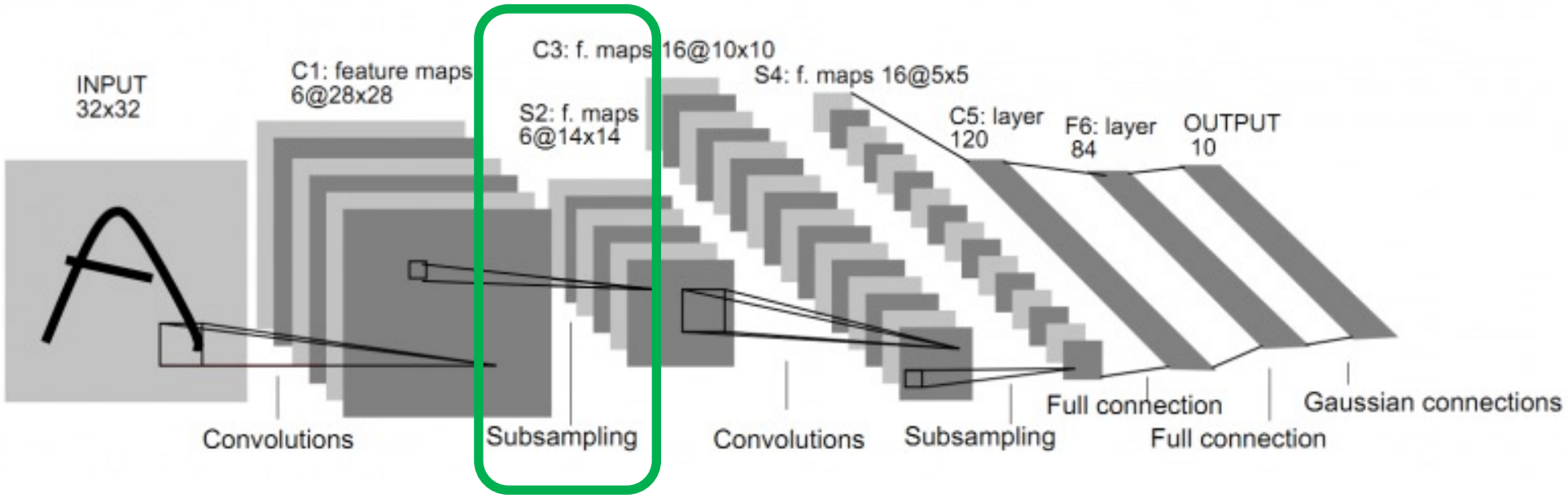
Convolutional layers



- Receptive field increases with increasing depth
  – Using small filters and more layers uses less parameters than large filters
- Feature maps shrink in size (if no padding is used)

What are we missing?

- First successful application of CNNs to a real-world problem: Handwritten digit recognition (LeCun et al, Proc. IEEE 1998)

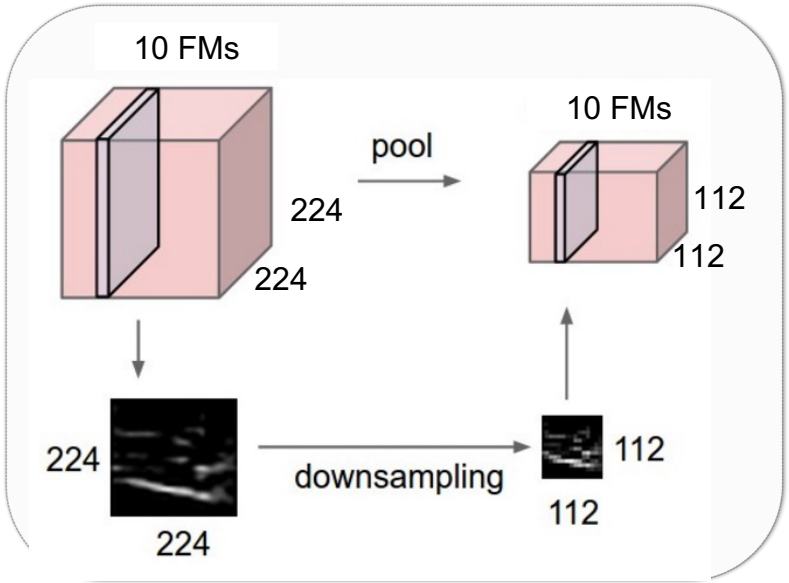


Subsampling or pooling layers

CNNs: Downsampling via Pooling

Motivation:

- Reduce size (memory) of deeper layers
- Invariance to small translations
- Contraction forces network to learn high-level features



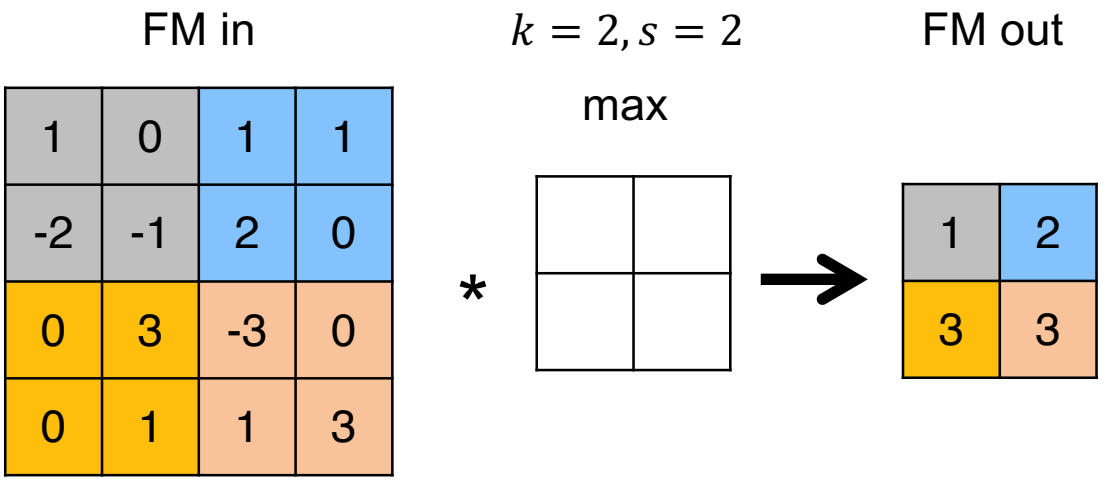
(Source: Stanford's CS231n github)

CNNs: Downsampling via Pooling

Pooling operator:

- Kernel of dimension k
- Stride of convolution s
- Applied to each FM
- Kernel function:
  - Max
  - Mean
  - …

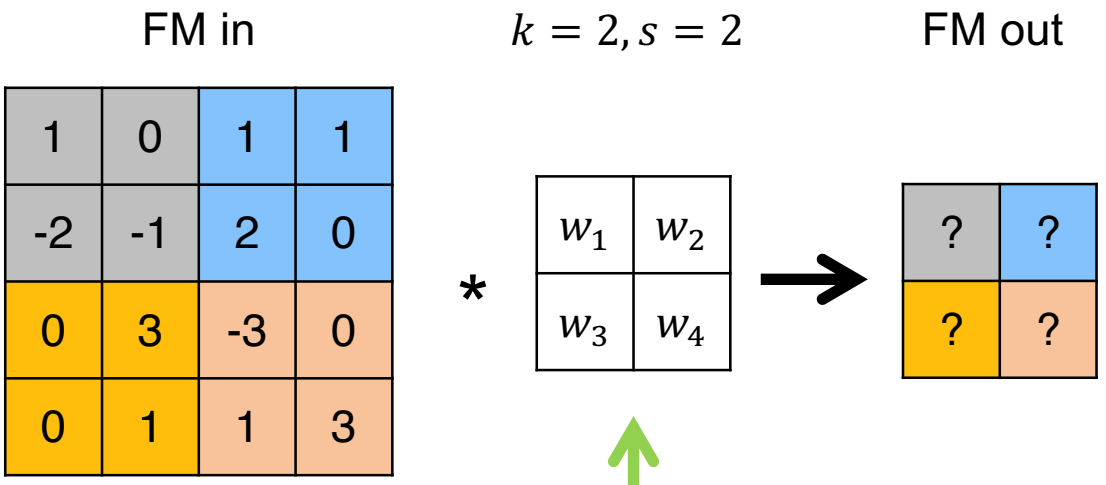
} No additional parameters



CNNs: Downsampling via Pooling

Pooling operator:

- Kernel of dimension k
- Stride of convolution s
- Applied to each FM
- Kernel function:
  - Max
  - Mean
  - …

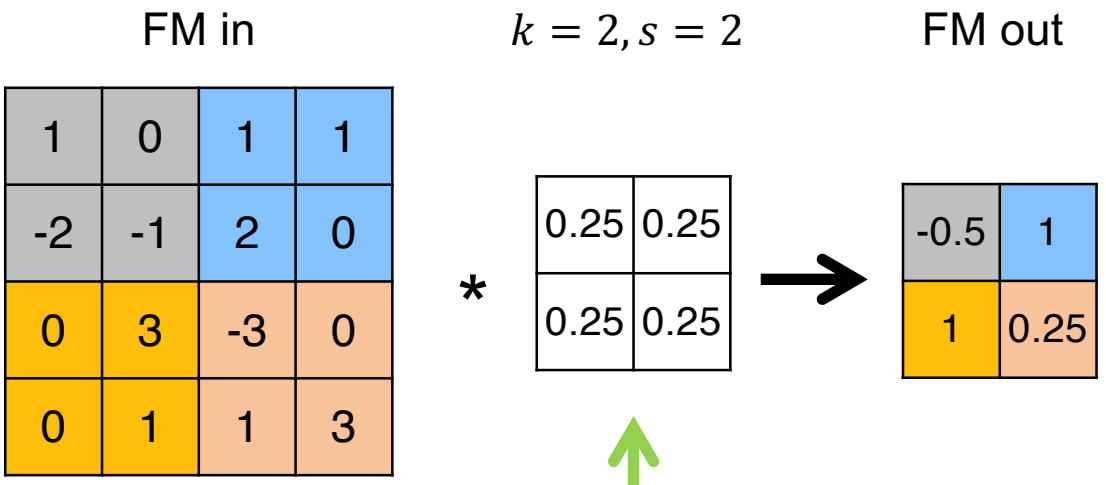


Additional parameters: Can be learnt

CNNs: Downsampling via Pooling

Pooling operator:

- Kernel of dimension k
- Stride of convolution s
- Applied to each FM
- Kernel function:
  - Max
  - Mean
  - …



Additional parameters: Can be learnt

CNNs: Upsampling

FM in

5	3	1
0	2	4
4	3	5

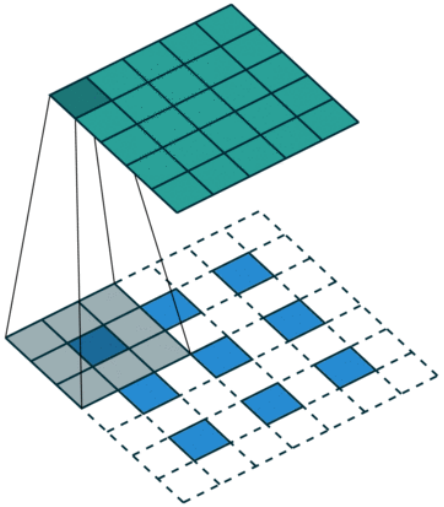
0	0	0	0	0	0	0	0
0	5	0	3	0	1	0	0
0	0	0	0	0	0	0	0
0	0	0	2	0	4	0	0
0	0	0	0	0	0	0	0
0	4	0	3	0	5	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

*

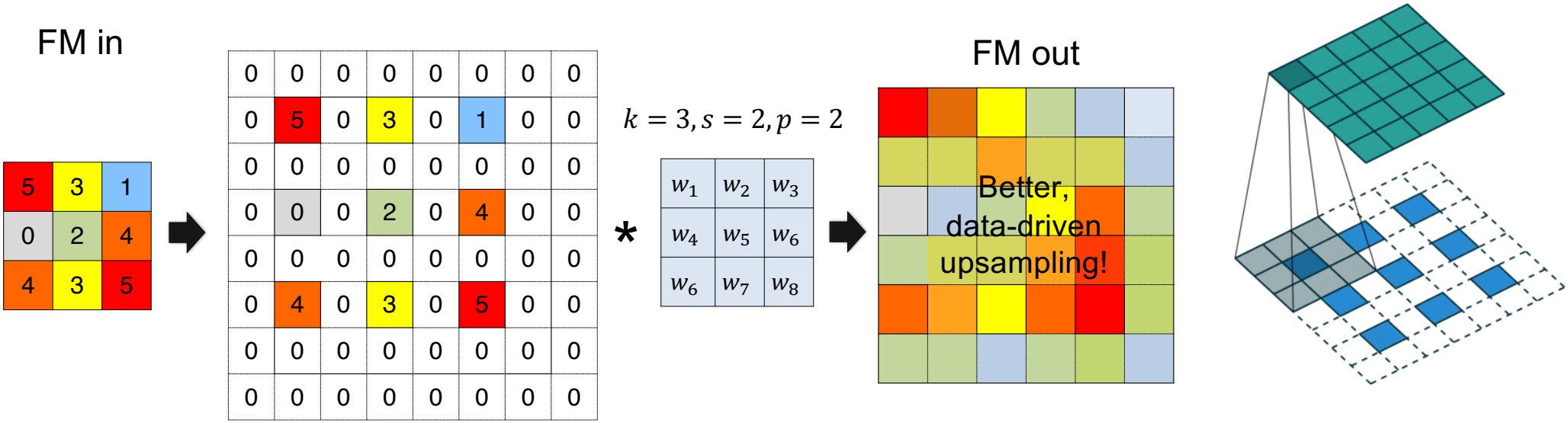
0.25	0.5	0.25
0.5	1	0.5
0.25	0.5	0.25

FM out

5	4	3	2	1	0.5
2.5	2.5	3.5	2.5	2.5	1.3
0	1	2	3	4	2
2	2.3	2.5	3.5	4.5	2.3
4	3.5	3	4	5	2.5
2	1.8	1.5	2	2.5	1.3

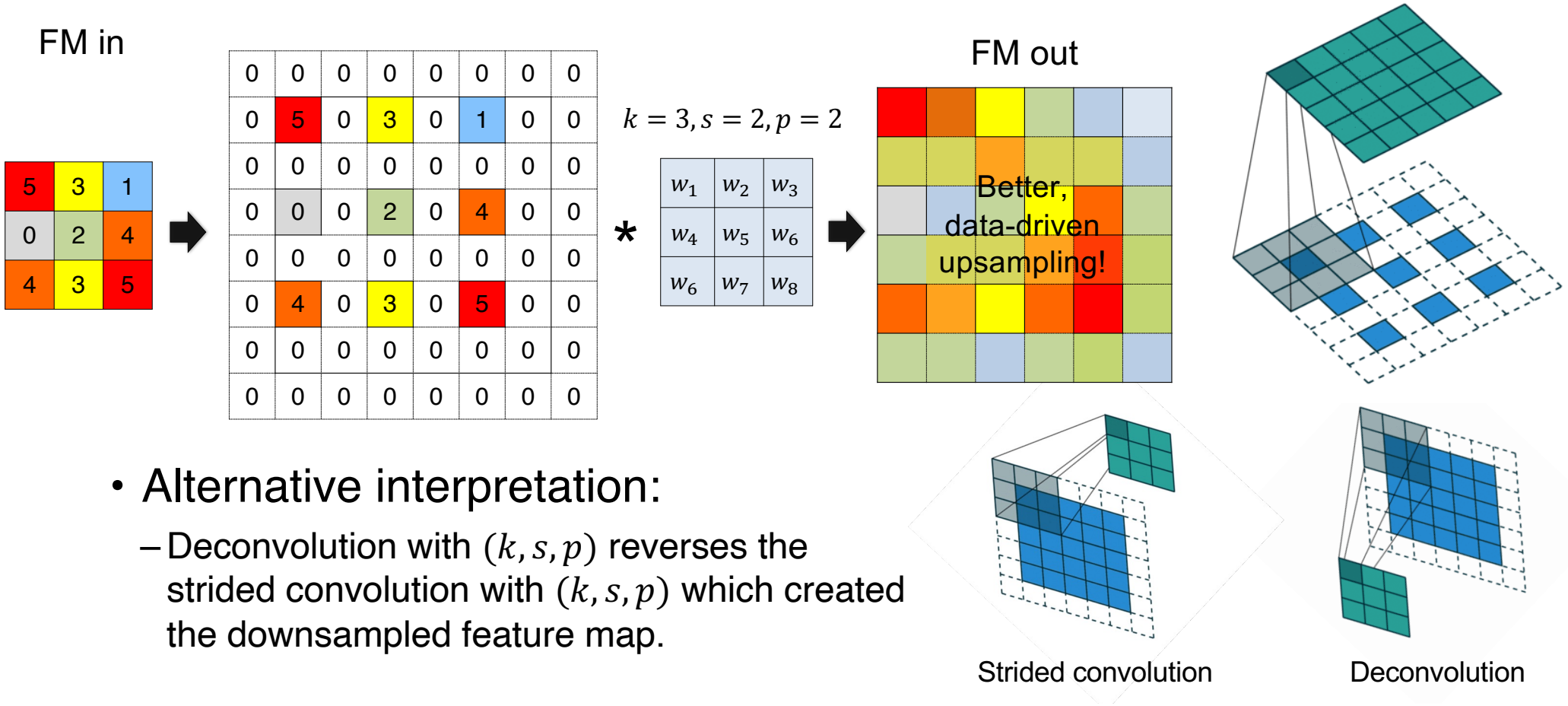


CNNs: Upsampling



- Deconvolution / transposed convolution: Learn to upsample
  - Kernel dimension k: how much context influences upsampling
  - Stride s of convolution: The upsampling factor
  - Padding p: careful, to get exact output shape wanted

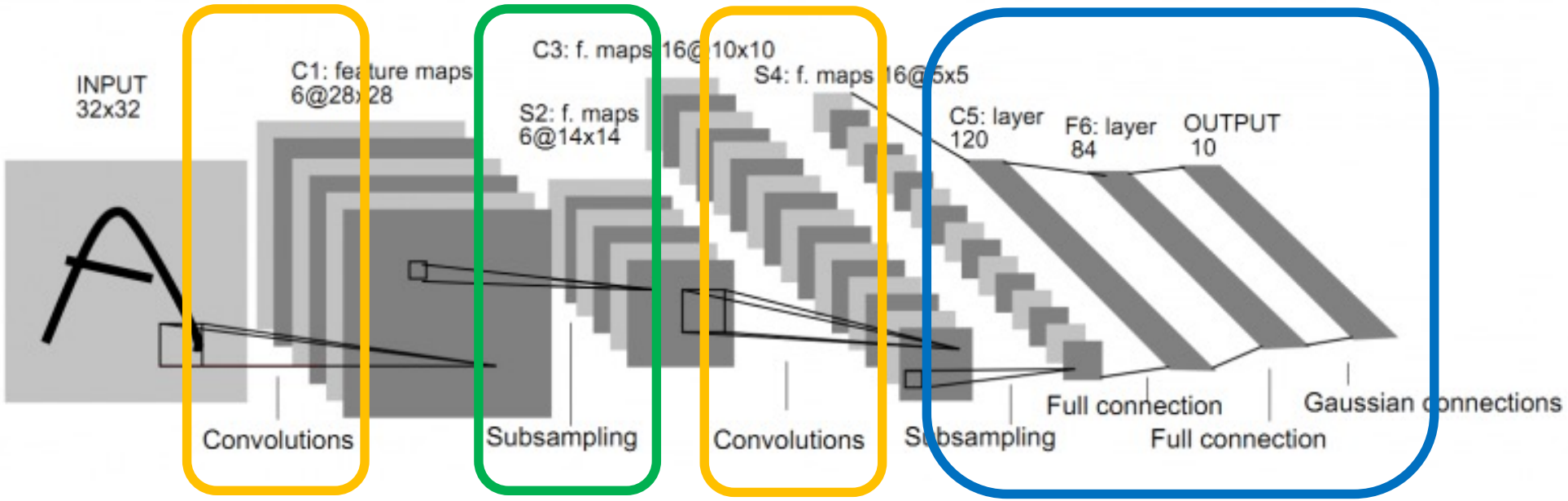
CNNs: Upsampling



- Alternative interpretation:
  - Deconvolution with (k, s, p) reverses the strided convolution with (k, s, p) which created the downsampled feature map.

LeNet

- First successful application of CNNs to a real-world problem: Handwritten digit recognition (LeCun et al, Proc. IEEE 1998)



Convolutional layers

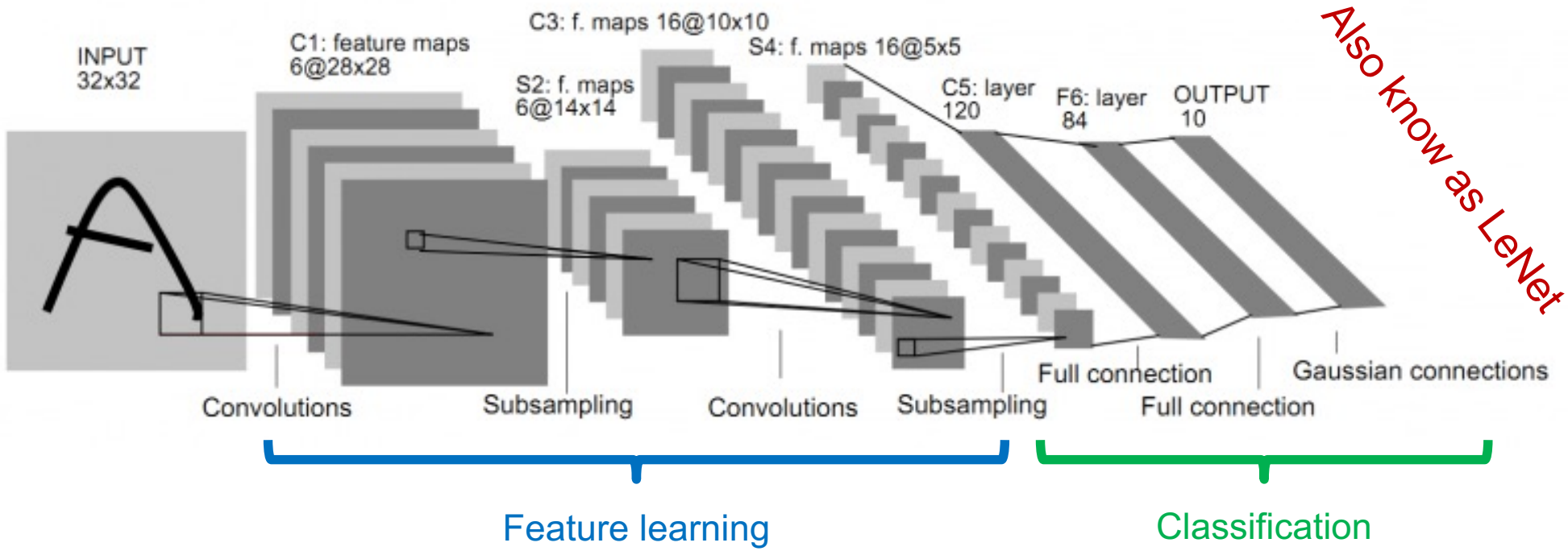
Subsampling or pooling layers

Convolutional layers

Fully connected layers

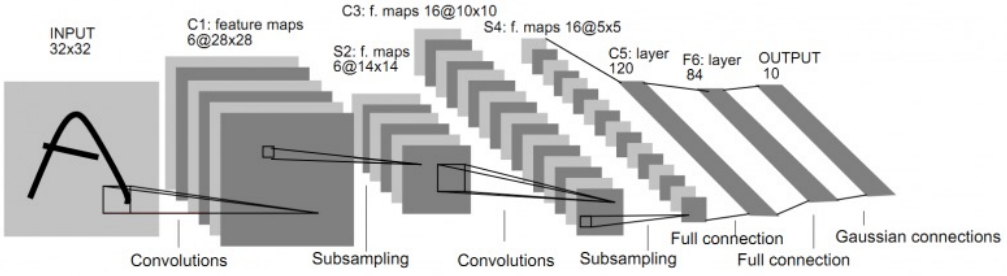
LeNet

- First successful application of CNNs to a real-world problem: Handwritten digit recognition (LeCun et al, Proc. IEEE 1998)



Putting it all together: LeNet

- First successful application of CNNs to a real-world problem: Handwritten digit recognition (LeCun et al, Proc. IEEE 1998)



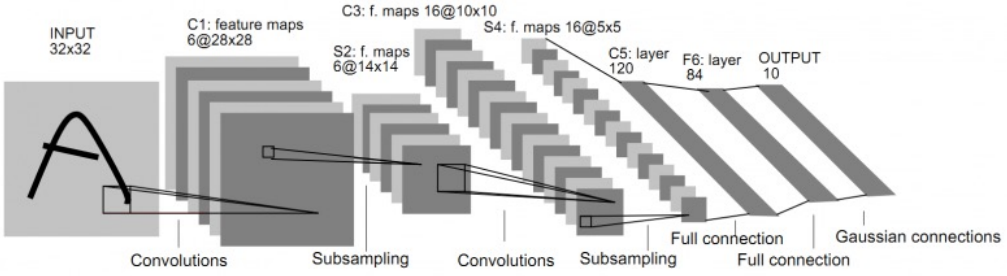
Also know as LeNet

5 x 5

Layer	Size	Parameters
Input	32 x 32	n/a
C1	28 x 28 x 6	156
S2	14 x 14 x 6	12
C3	10 x 10 x 16	1516
S4	5 x 5 x 16	32
C5	120	48120
F6	84	10164
Output	10 x 1	n/a

Putting it all together: LeNet

- First successful application of CNNs to a real-world problem: Handwritten digit recognition (LeCun et al, Proc. IEEE 1998)



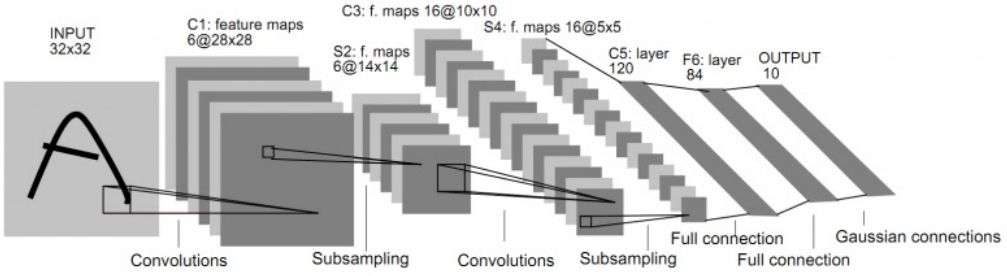
Also know as LeNet

trainable multiplier and offset

Layer	Size	Parameters
Input	32 x 32	n/a
C1	28 x 28 x 6	156
S2	14 x 14 x 6	12
C3	10 x 10 x 16	1516
S4	5 x 5 x 16	32
C5	120	48120
F6	84	10164
Output	10 x 1	n/a

Putting it all together: LeNet

- First successful application of CNNs to a real-world problem: Handwritten digit recognition (LeCun et al, Proc. IEEE 1998)



Also know as LeNet

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	X				X	X	X			X	X	X	X		X	X
1	X	X				X	X	X			X	X	X	X		X
2	X	X	X				X	X	X			X		X	X	X
3		X	X	X			X	X	X	X			X		X	X
4			X	X	X			X	X	X	X		X	X		X
5				X	X	X			X	X	X	X		X	X	X

TABLE I

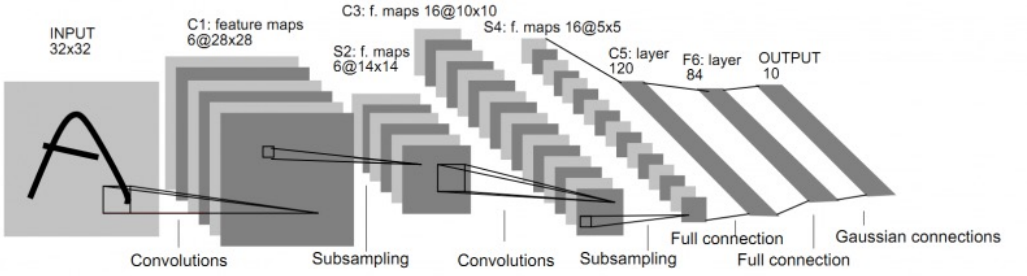
EACH COLUMN INDICATES WHICH FEATURE MAP IN S2 ARE COMBINED BY THE UNITS IN A PARTICULAR FEATURE MAP OF C3.

Layer	Size	Parameters
Input	32 x 32	n/a
C1	28 x 28 x 6	156
S2	14 x 14 x 6	12
C3	10 x 10 x 16	1516
S4	5 x 5 x 16	32
C5	120	48120
F6	84	10164
Output	10 x 1	n/a

Only 60k parameters

Putting it all together: LeNet

- First successful application of CNNs to a real-world problem: Handwritten digit recognition (LeCun et al, Proc. IEEE 1998)



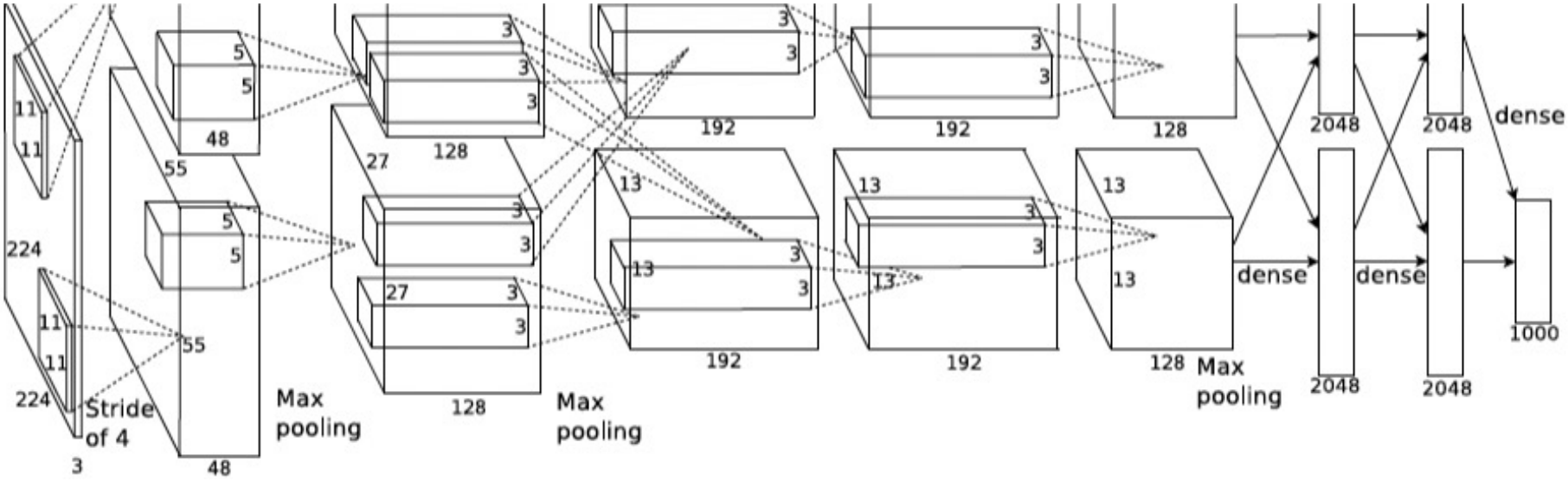
Also know as LeNet

– Convolution for spatial features
– Subsampling via average pooling
– Activation function: tanh
– Sparse connectivity between S2 and C3

– MLP as final classifier
– Sequence: Convolution, Subsampling, Non-linearity
– Inspired many of the architectures on the next slides

AlexNet

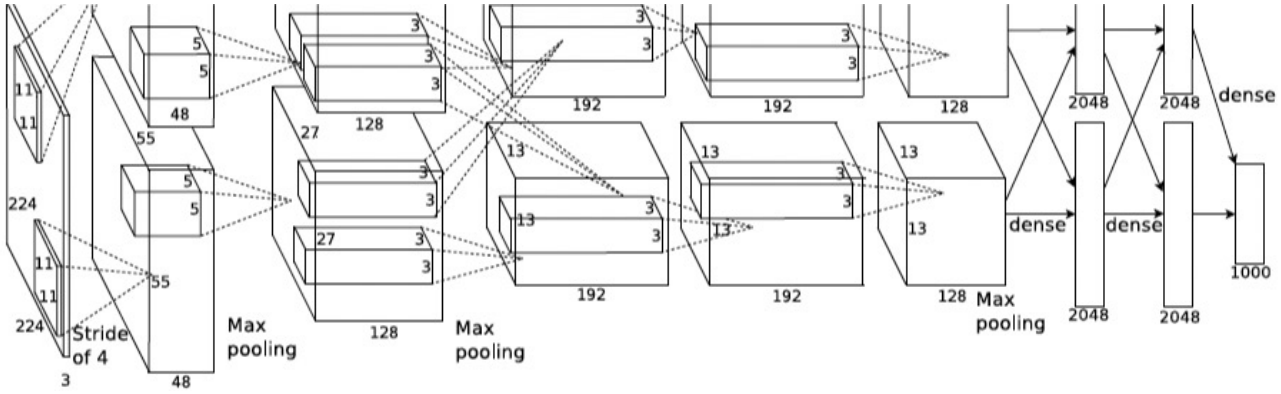
- Winner of the the ImageNet 2012 Challenge (A. Krizhevsky et al. NIPS* 2012)



*since 2018 NIPS is called NeurIPS

AlexNet

- Similar to LeNet, but:
  - Max-pooling and ReLU
  - Stacked convolutional layers directly on top of each other, instead of stacking a pooling layer on top of each convolutional layer
  - Dropout regularization
  - Uses data augmentation (random transformation, random intensity variation)
  - Trained on two GPUs for a week



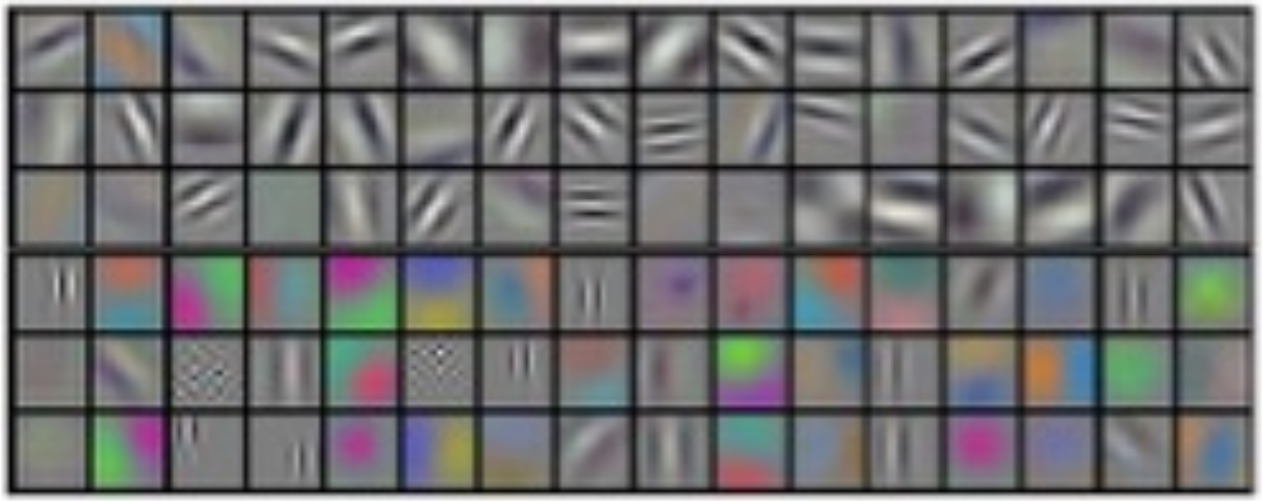
AlexNet

Layer	Type	Maps	Size	Kernel size	Stride	Padding	Activation
Out	Fully Connected	–	1,000	–	–	–	Softmax
F9	Fully Connected	–	4,096	–	–	–	ReLU
F8	Fully Connected	–	4,096	–	–	–	ReLU
C7	Convolution	256	13 × 13	3 × 3	1	SAME	ReLU
C6	Convolution	384	13 × 13	3 × 3	1	SAME	ReLU
C5	Convolution	384	13 × 13	3 × 3	1	SAME	ReLU
S4	Max Pooling	256	13 × 13	3 × 3	2	VALID	–
C3	Convolution	256	27 × 27	5 × 5	1	SAME	ReLU
S2	Max Pooling	96	27 × 27	3 × 3	2	VALID	–
C1	Convolution	96	55 × 55	11 × 11	4	SAME	ReLU
In	Input	3 (RGB)	224 × 224	–	–	–	–

Bigger model (7 hidden layers, 650K units, 60M parameters)

AlexNet

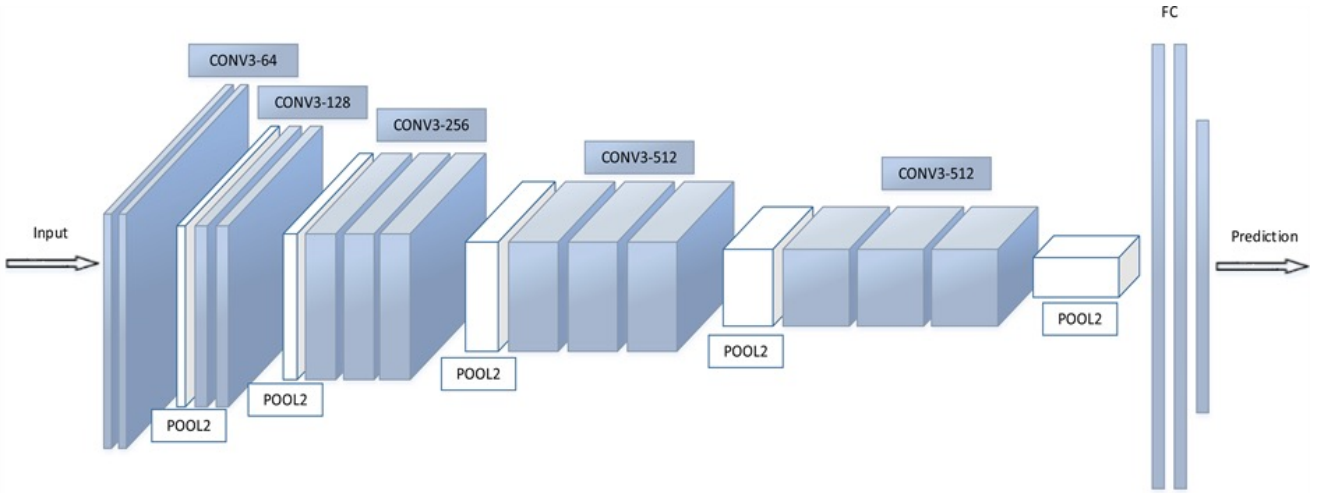
- 96 convolutional kernels of size 11 × 11 × 3 learnt by first convolutional layer on the 224 × 224 × 3 input images.



– Used two GPUs, splitting kernels into 48/48

VGGNet

- Very Deep Convolutional Networks for Large-Scale Visual Recognition (K. Simonyan and A. Zisserman, arXiv, 2014)

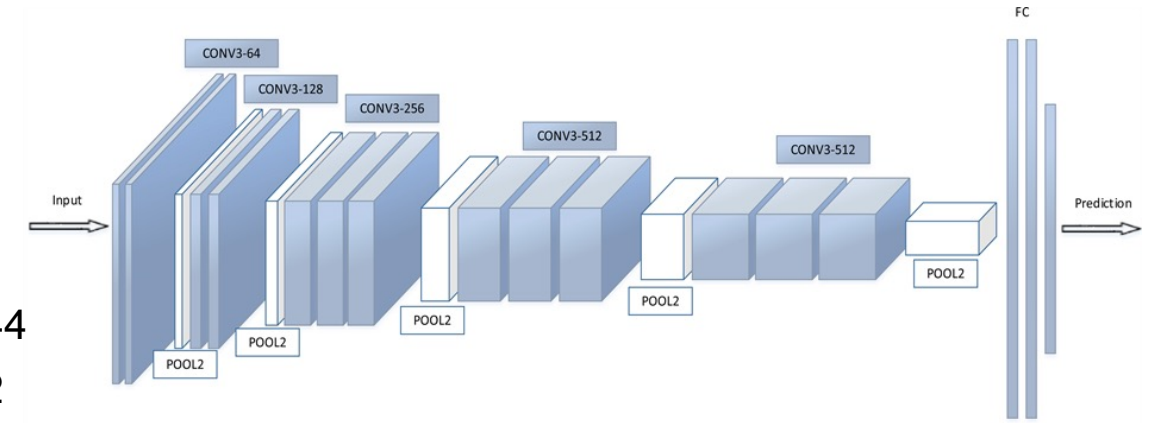


  - Runner-up ILSVRC-2014
  - Rigorous evaluation of networks of increasing depth, up to 16-19 weight layers
  - Very small 3×3 filters in all convolutional layers (the convolution stride is set to 1).
  - Trained on 4 GPUs for 2–3 weeks

VGGNet

- Very Deep Convolutional Networks for Large-Scale Visual Recognition (K. Simonyan and A. Zisserman, arXiv, 2014)
- Number of parameters:
  - conv3-64 x 2: 38,720
  - conv3-128 x 2: 221,440
  - conv3-256 x 3: 1,475,328
  - conv3-512 x 3: 5,899,776
  - conv3-512 x 3: 7,079,424
  - fc1: 102,764,544
  - fc2: 16,781,312
  - fc3: 4,097,000

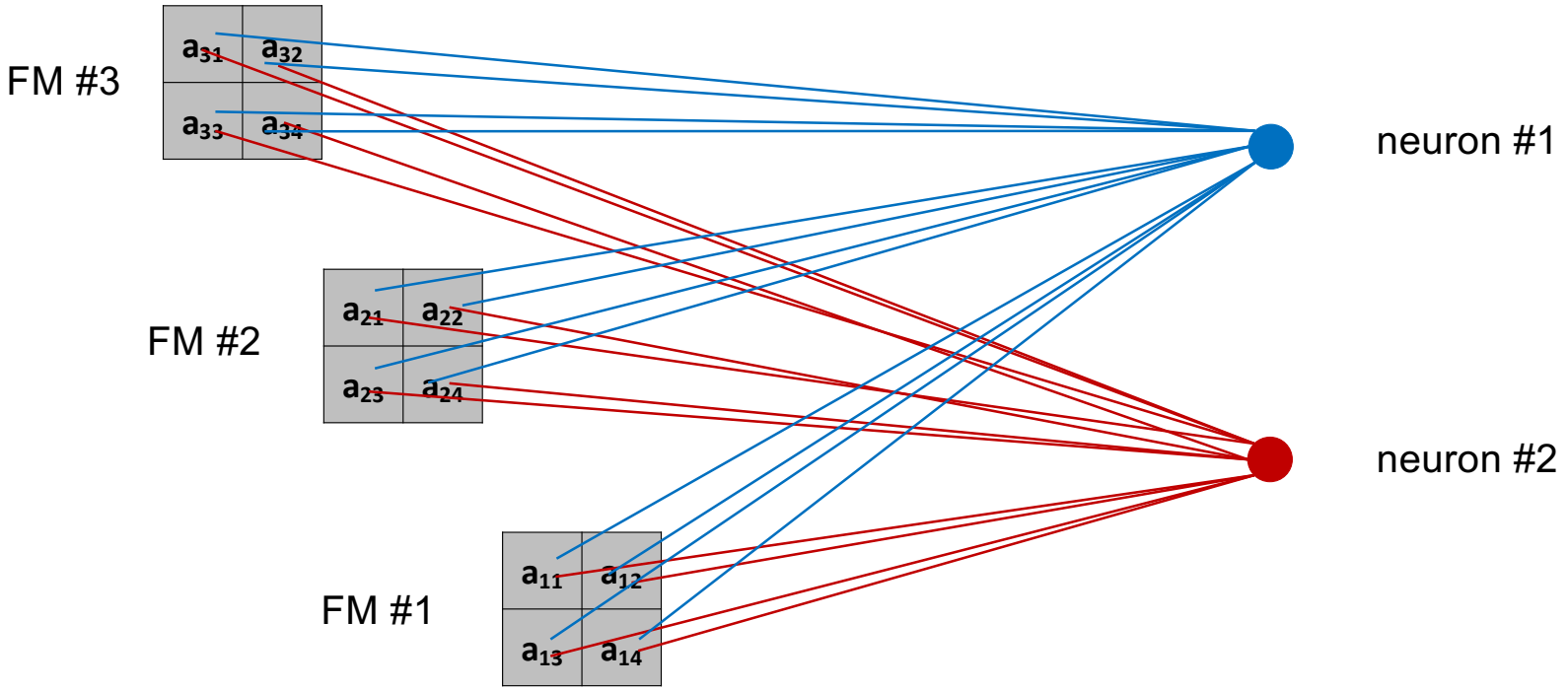
  - **TOTAL:** **138,357,544**



Changing fully connected layers to convolutional layers

3 x 2D feature maps (FM) of previous convolutional layer

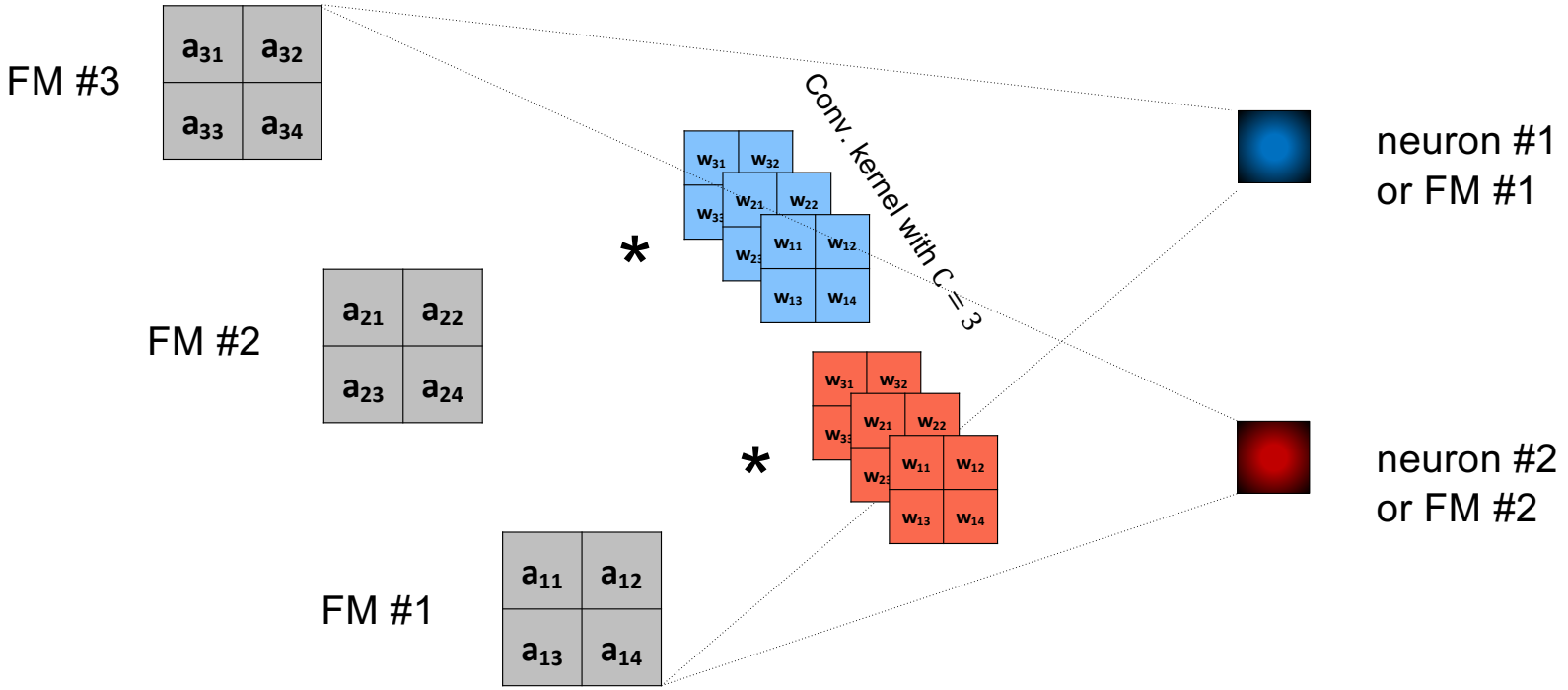
Fully connected neurons of next fully connected layer



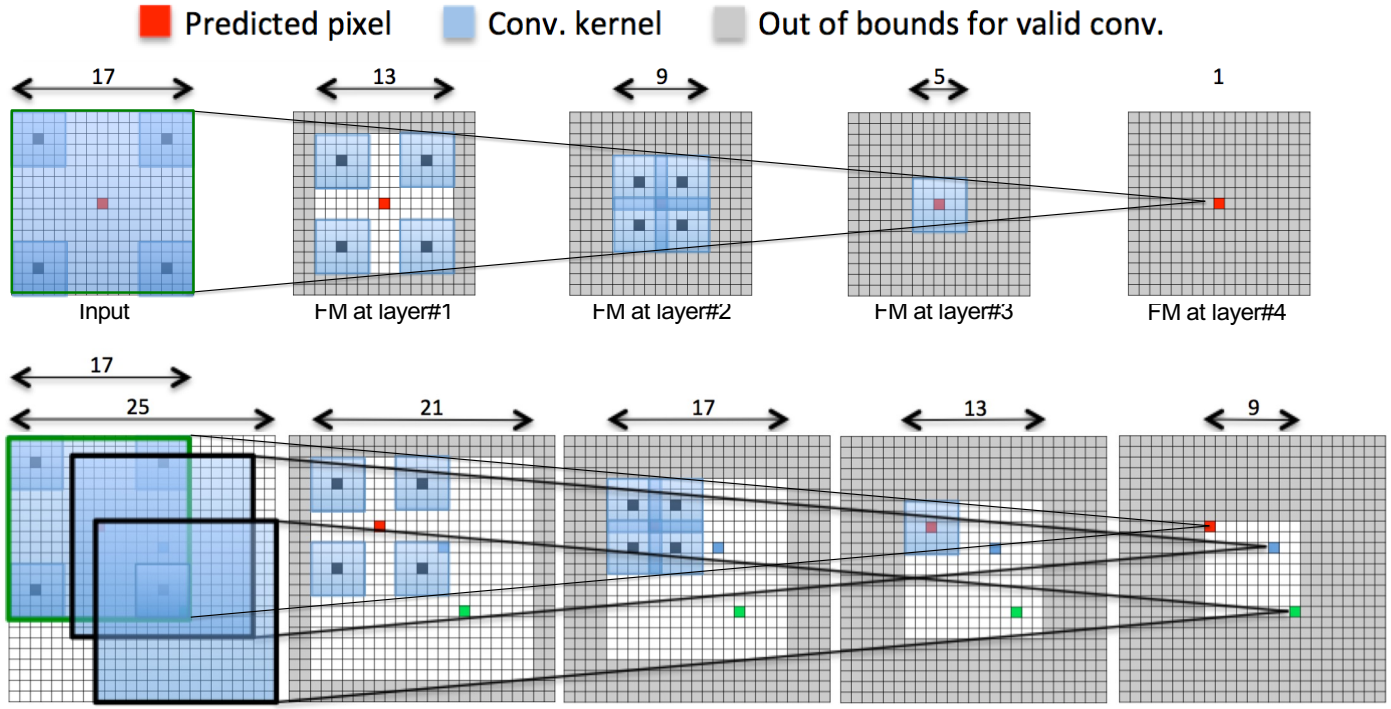
Changing fully connected layers to convolutional layers

3 x 2D feature maps (FM) of previous convolutional layer

Fully connected neurons of next fully connected layer



Fully convolutional networks: Size of feature maps expands with input



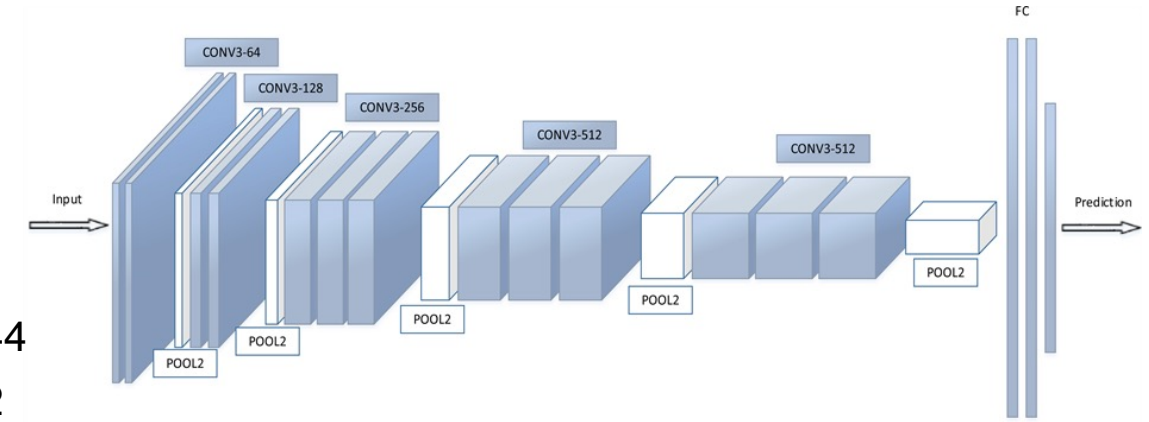
- **Size of receptive field** remains the same!
- Each prediction is only influenced by its own receptive field. NOT the whole input!

Problems with CNN architectures so far

- CNNs can get very large, e.g. VGG

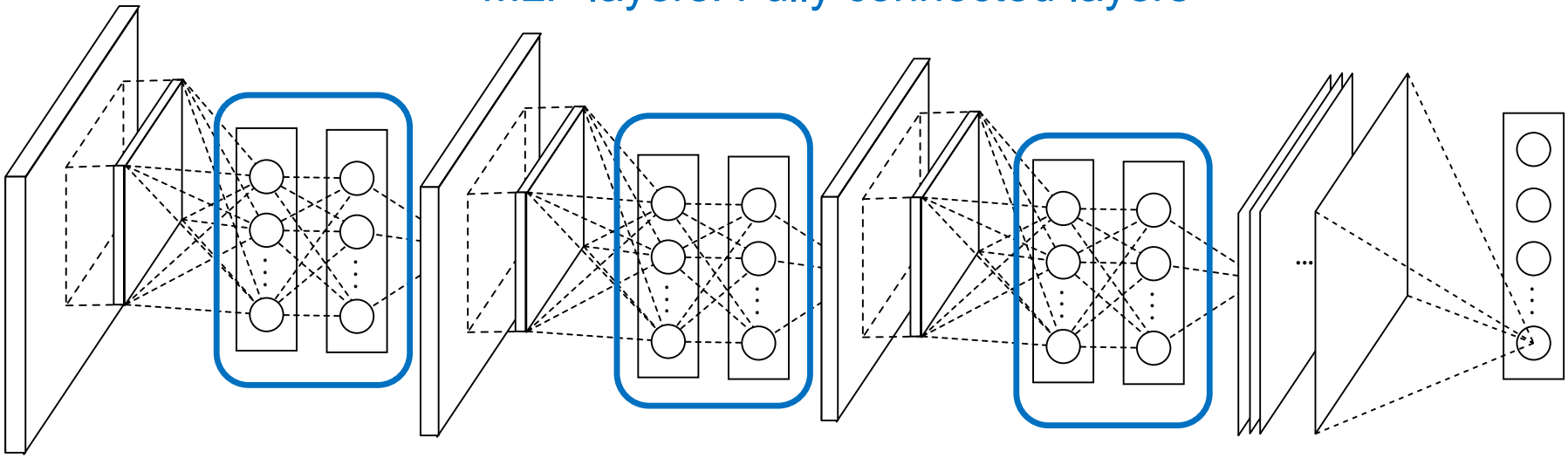
- Number of parameters:
  - conv3-64 x 2: 38,720
  - conv3-128 x 2: 221,440
  - conv3-256 x 3: 1,475,328
  - conv3-512 x 3: 5,899,776
  - conv3-512 x 3: 7,079,424
  - fc1: 102,764,544
  - fc2: 16,781,312
  - fc3: 4,097,000

  - **TOTAL:** **<u>138,357,544</u>**



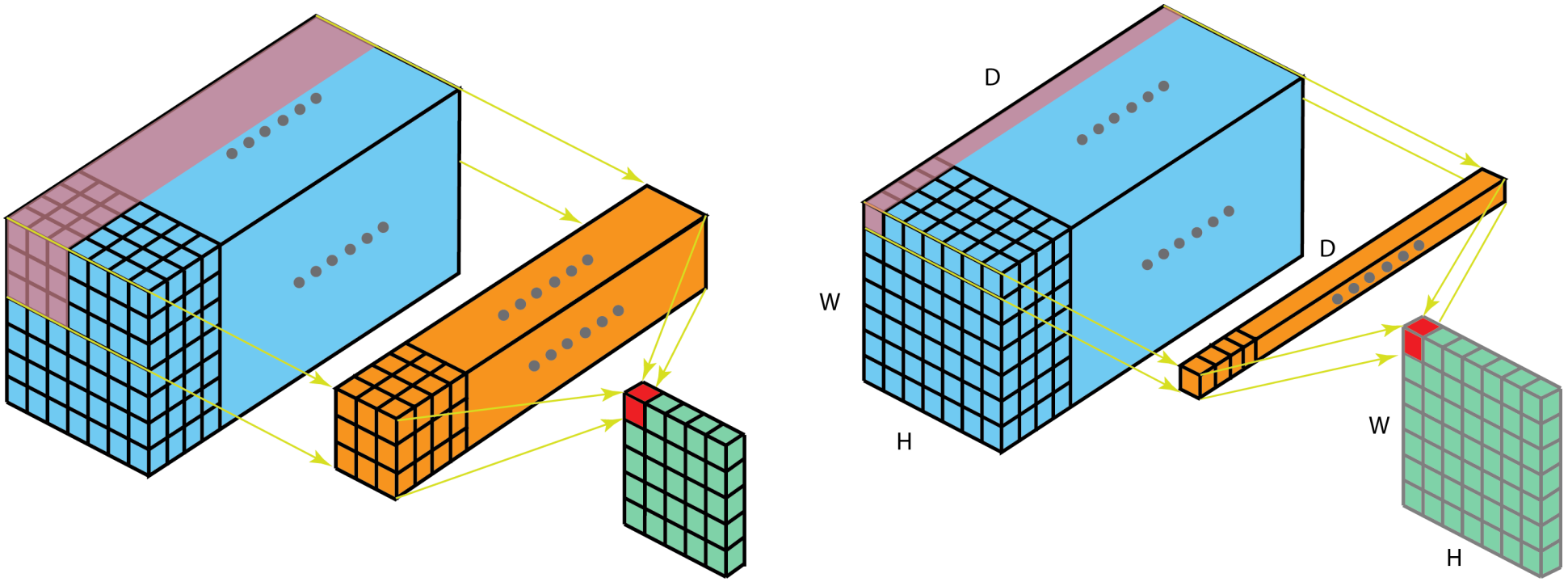
Network in network

MLP layers: Fully connected layers



(M. Lin et al., arXiv, 2013)

Use 1 x 1 convolutions

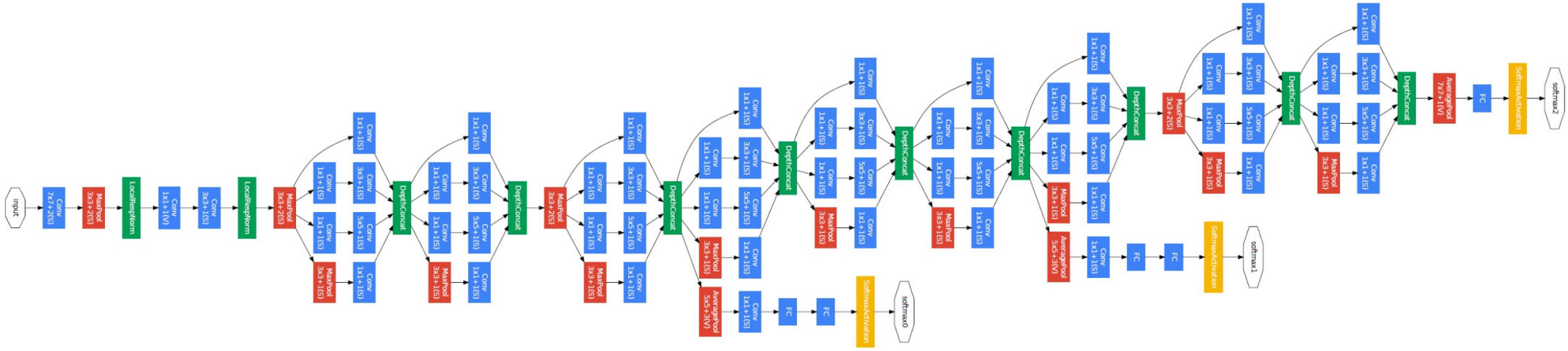


https://towardsdatascience.com/a-comprehensive-introduction-to-different-types-of-convolutions-in-deep-learning-669281e58215

The need for depth

GoogLeNet (aka Inception V1)

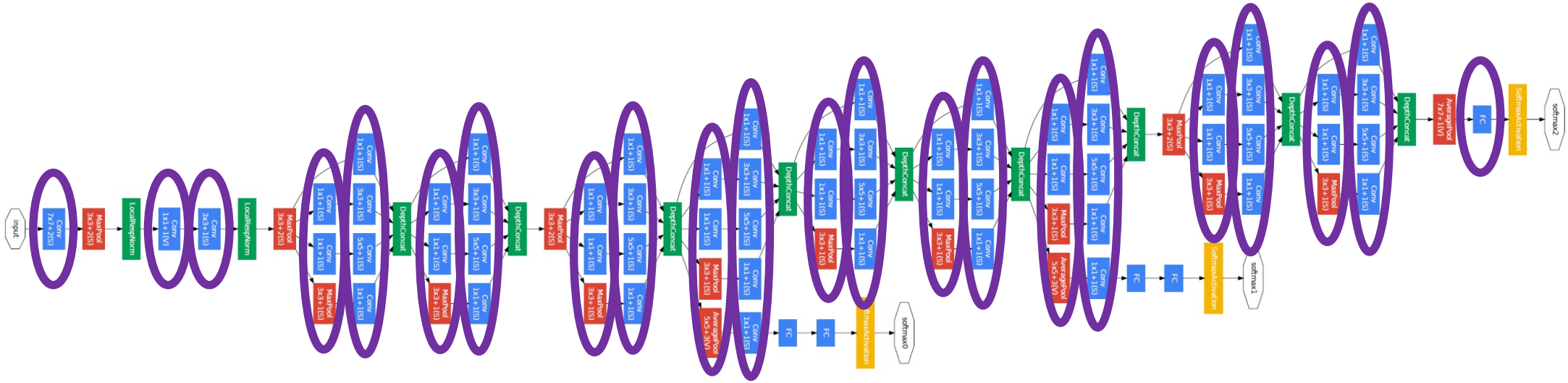
- Going Deeper with Convolutions (C. Szegedy et al., CVPR 2015)



– Winner of the ILSVRC 2014 competition (close to human level performance)

GoogLeNet (aka Inception V1)

- Going Deeper with Convolutions (C. Szegedy et al., CVPR 2015)

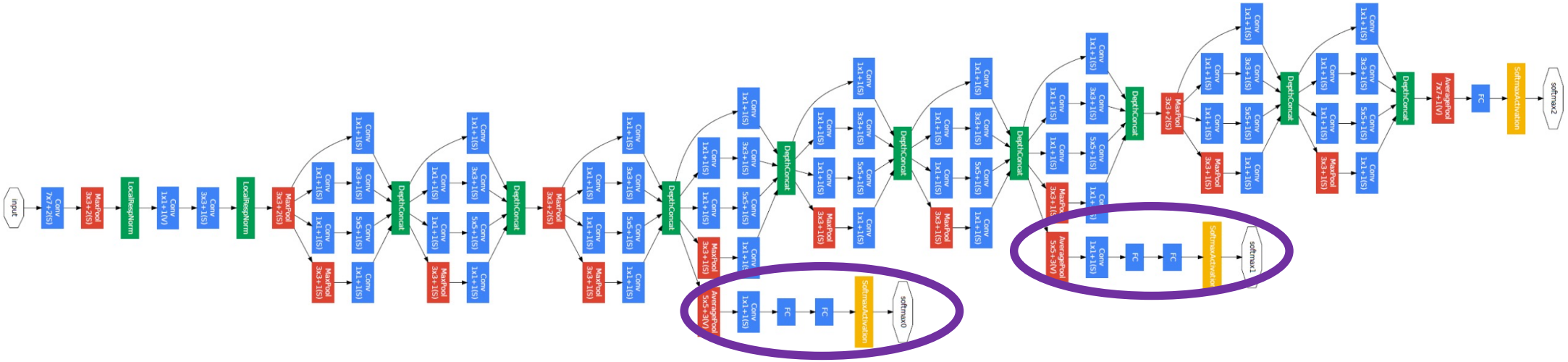


– Winner of the ILSVRC 2014 competition (close to human level performance)

– 22 layers + global average pooling as final layer

GoogLeNet (aka Inception V1)

- Going Deeper with Convolutions (C. Szegedy et al., CVPR 2015)



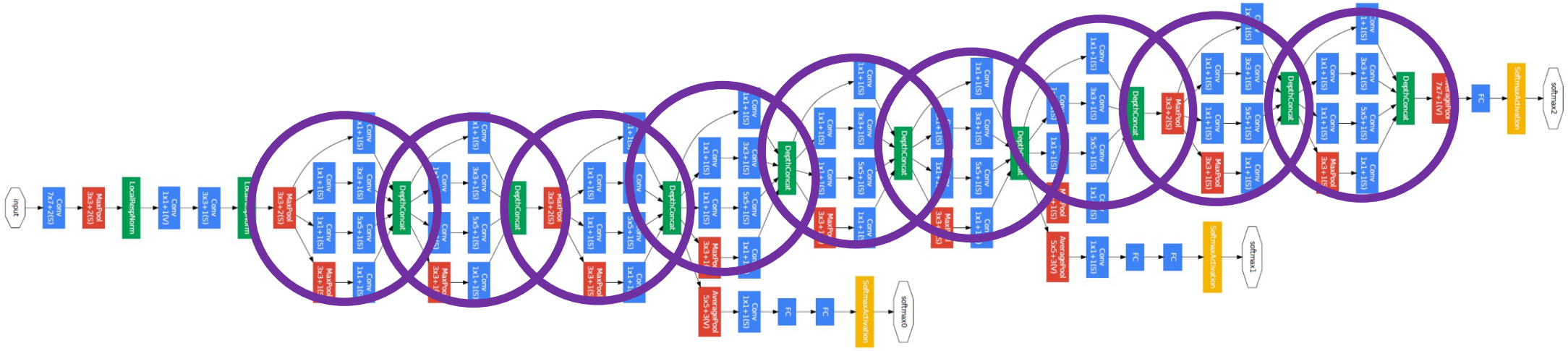
– Winner of the ILSVRC 2014 competition (close to human level performance)

– 22 layers + global average pooling as final layer

– Auxiliary classifiers (only used during training)

GoogLeNet (aka Inception V1)

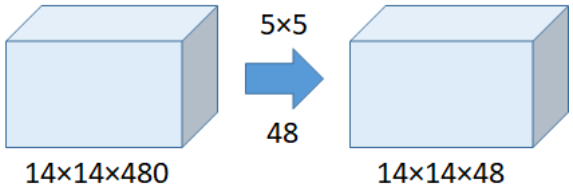
- Going Deeper with Convolutions (C. Szegedy et al., CVPR 2015)



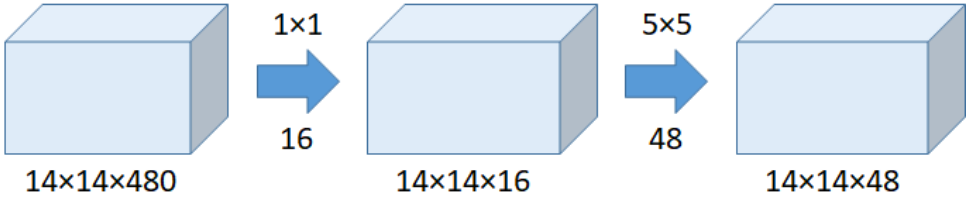
– Winner of the ILSVRC 2014 competition (close to human level performance)
– 22 layers + global average pooling as final layer
– Auxiliary classifiers (only used during training)
– Inception modules

GoogLeNet (aka Inception V1)

- Makes extensive use of 1 x 1 convolutions
  - Without 1 x 1 convolutions:

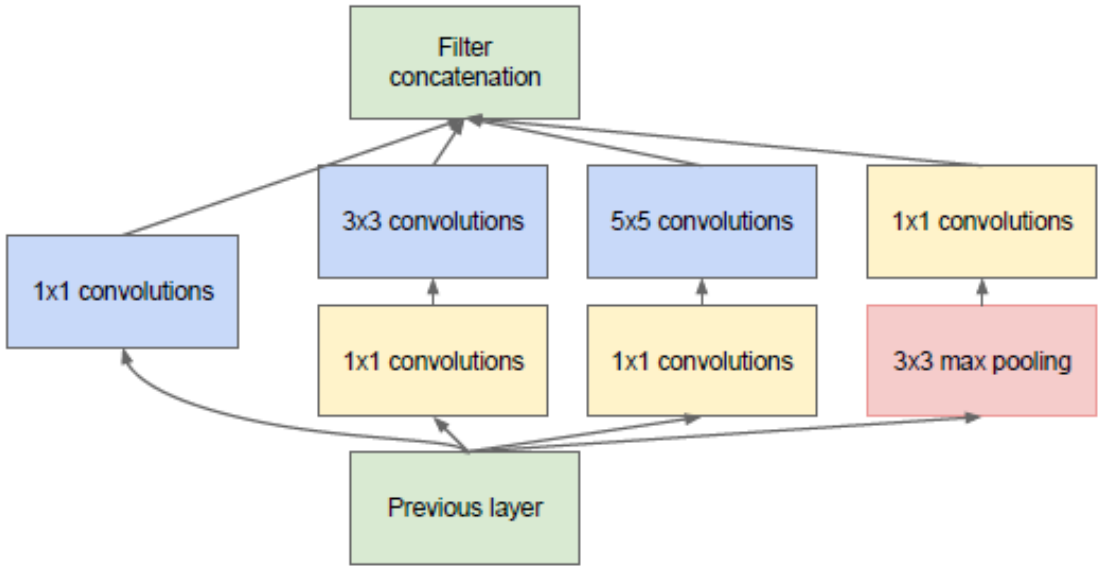


    - **Number of operations: 112.9M**
  - With 1 x 1 convolutions:



    - Number of operations for 1×1 = (14×14×16)×(1×1×480) = 1.5M
    - Number of operations for 5×5 = (14×14×48)×(5×5×16) = 3.8M
    - **Total number of operations = 1.5M + 3.8M = 5.3M**

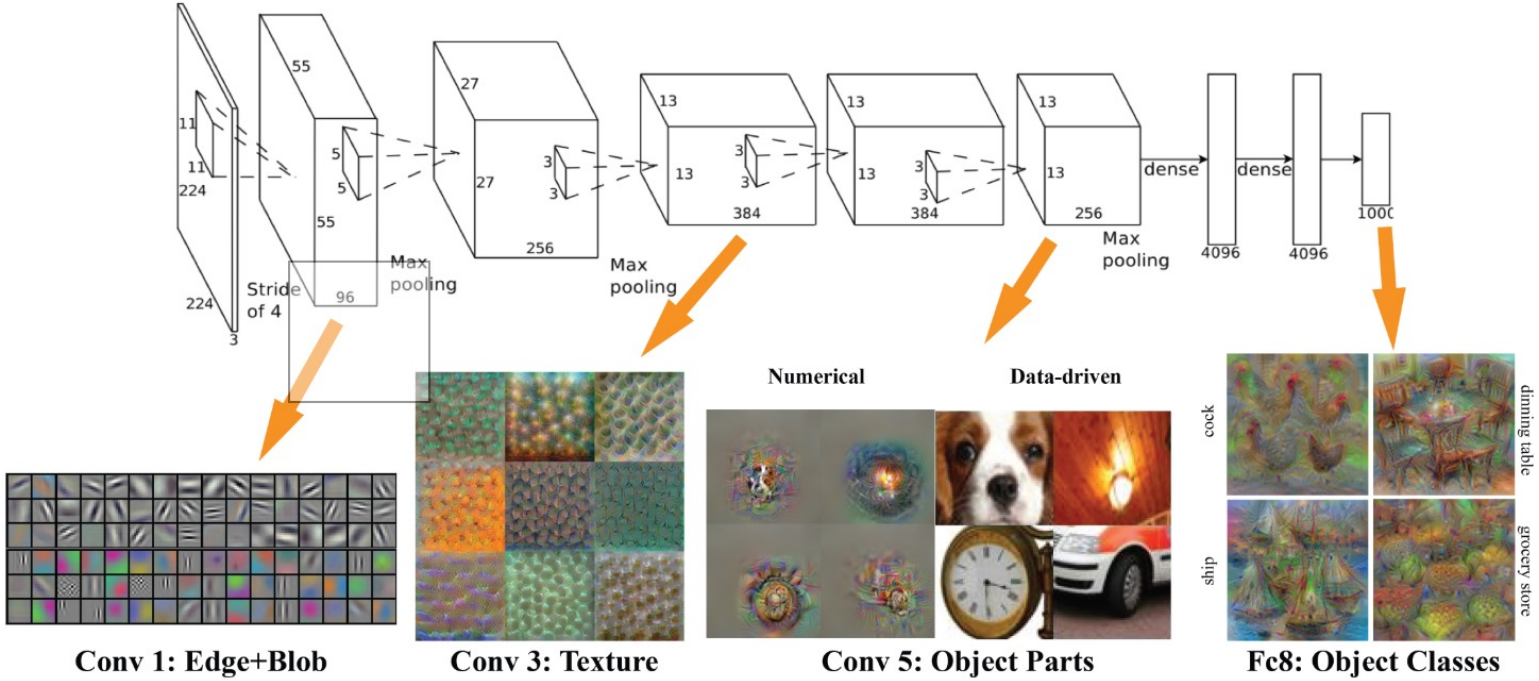
GoogLeNet (aka Inception V1)



– Derived from NiN concept (see previous slides)
– Parallel filter combinations (split-transform-merge strategy)
  • Network *learns* optimal filter size
– 1 x 1 convolution layers serve as “bottleneck” layers
– Representational power of large and dense layers with significantly fewer parameters and reduced complexity

Deeper models

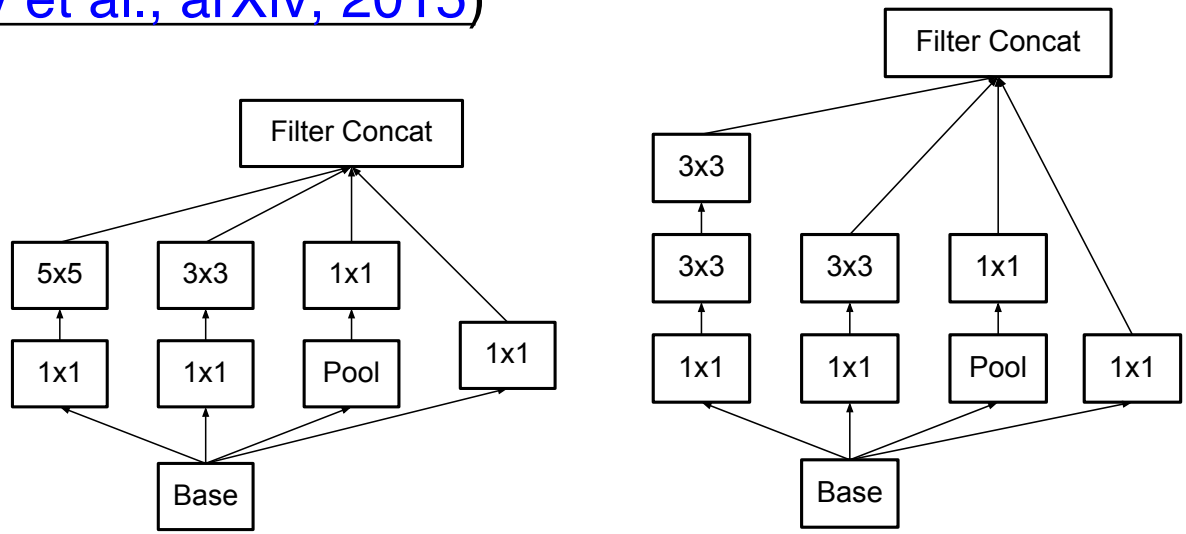
- With increasing depth features get increasingly abstract



http://vision03.csail.mit.edu/cnn_art/

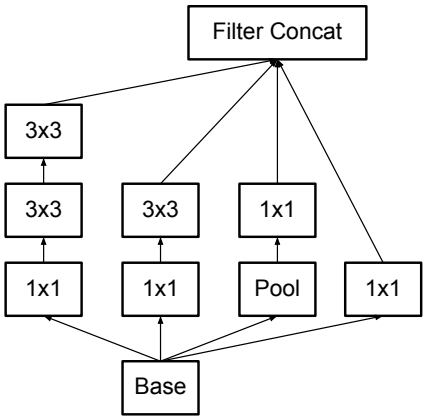
Inception V2

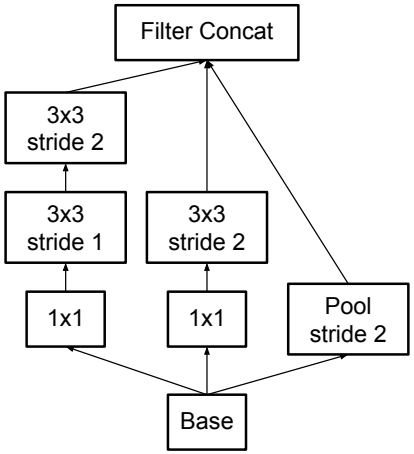
- Rethinking the Inception Architecture for Computer Vision (C. Szegedy et al., arXiv, 2015)

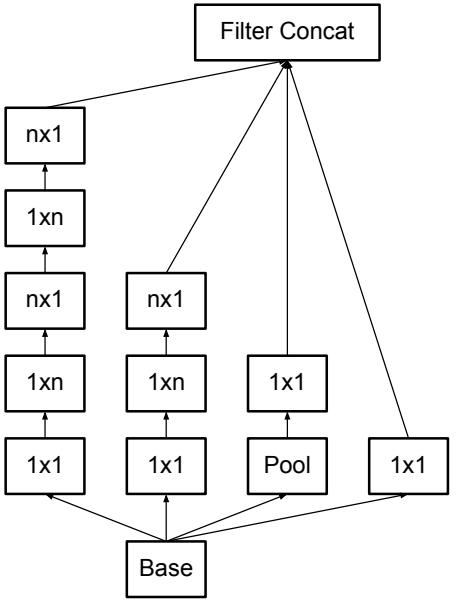


– Change basic inception layer: Replace 7 x 7 and 5 x 5 filters with multiple 3 x 3 convolutions

Inception V2





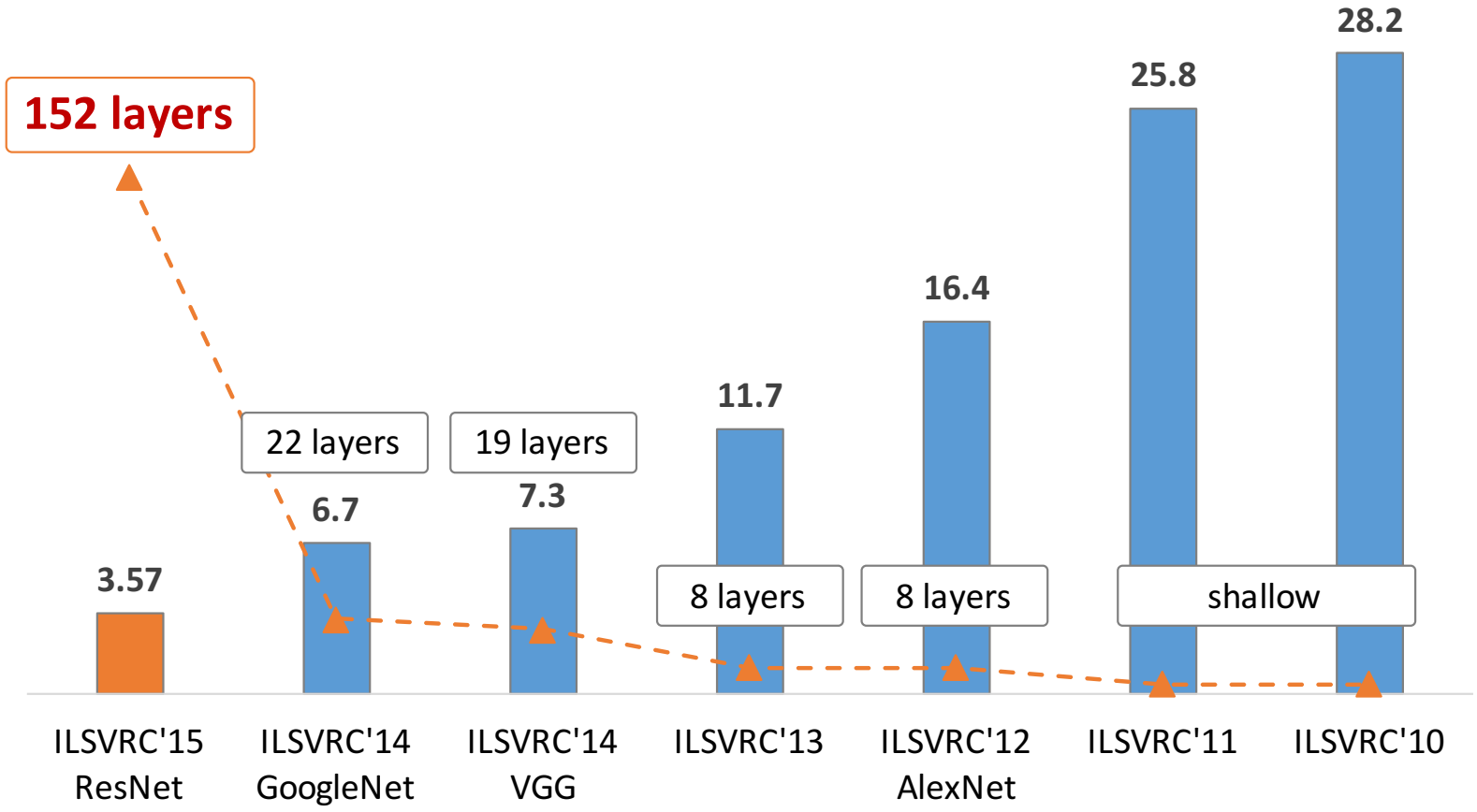


– 42 layers: Start with several 3 x 3 convolutions and 3 modified inception modules
– Efficient grid size reduction
– 5 modules of flattended convolutions (n = 7)
– Efficient grid size reduction + average pooling + softmax

Inception V3

- Uses
  - RMSProp Optimizer
  - Factorized 7x7 convolutions
- BatchNorm in the Auxillary Classifiers.
- Label Smoothing
  - Regularization added to the loss formula to deal with label noise
  - prevents the network from becoming too confident about a class. Prevents overfitting.

Going even deeper

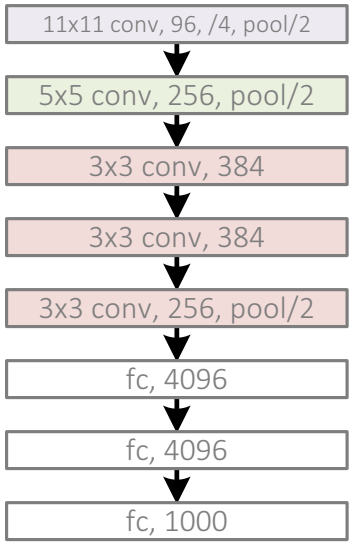


ImageNet Classification top-5 error (%)

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

Going even deeper

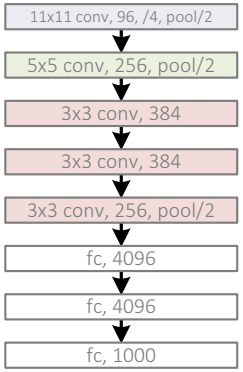
AlexNet, 8 layers
(ILSVRC 2012)



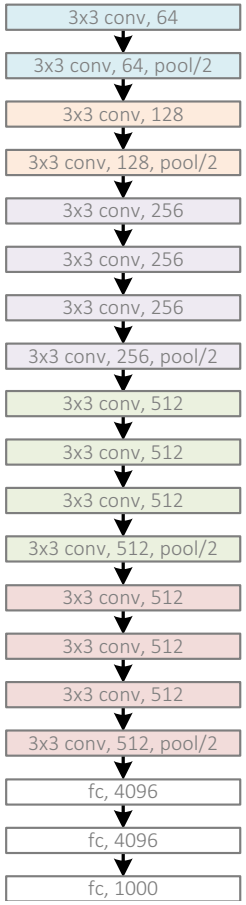
Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. “Deep Residual Learning for Image Recognition”. CVPR 2016.

Going even deeper

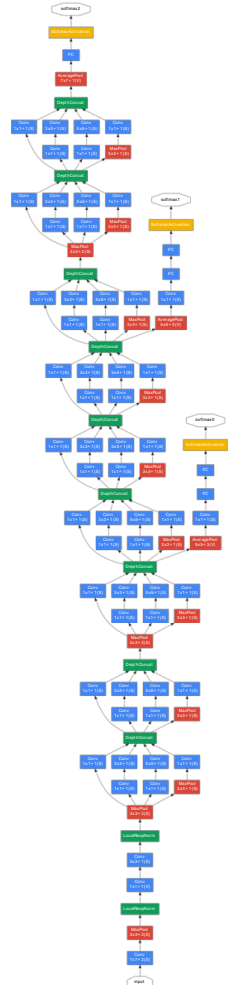
AlexNet, 8 layers
(ILSVRC 2012)



VGG, 19 layers
(ILSVRC 2014)



GoogleNet, 22 layers
(ILSVRC 2014)



Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. “Deep Residual Learning for Image Recognition”. CVPR 2016.

Going even deeper

AlexNet, 8 layers
(ILSVRC 2012)

VGG, 19 layers
(ILSVRC 2014)

ResNet, 152 layers
(ILSVRC 2015)

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

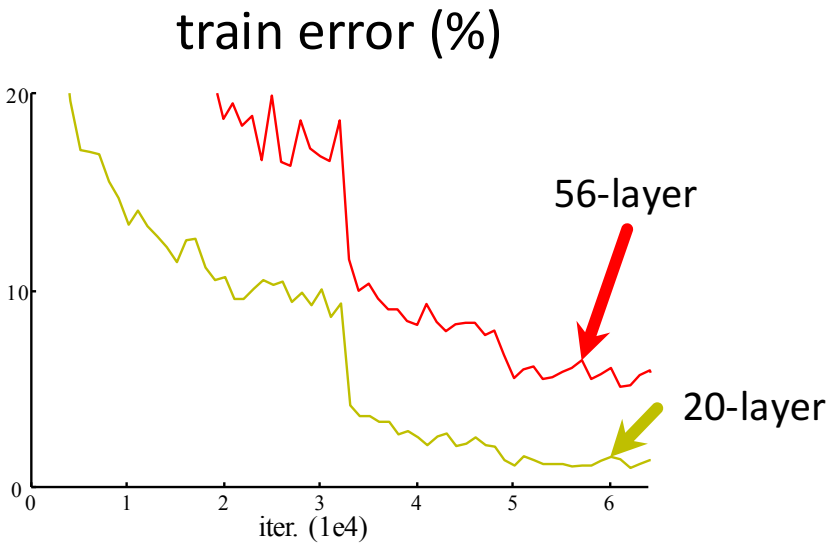
Going even deeper

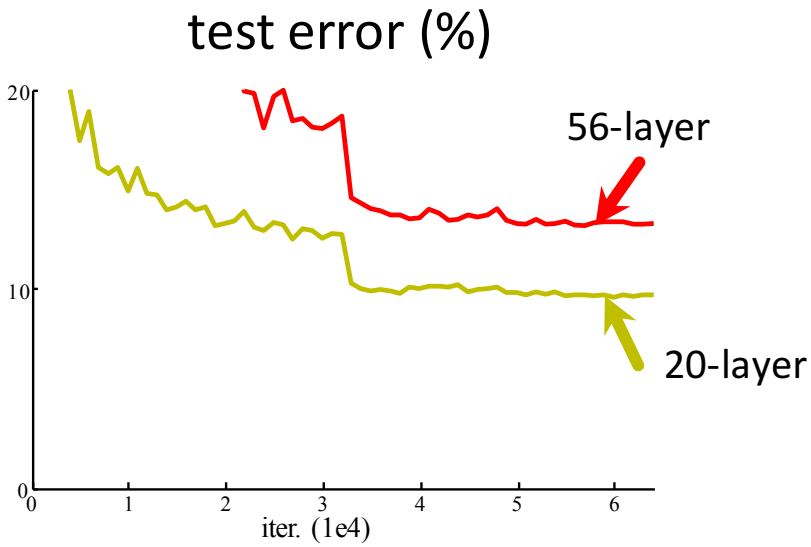
- Is learning better networks as simple as stacking more layers?

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. “Deep Residual Learning for Image Recognition”. CVPR 2016.

Going even deeper

- Is learning better networks as simple as stacking more layers?

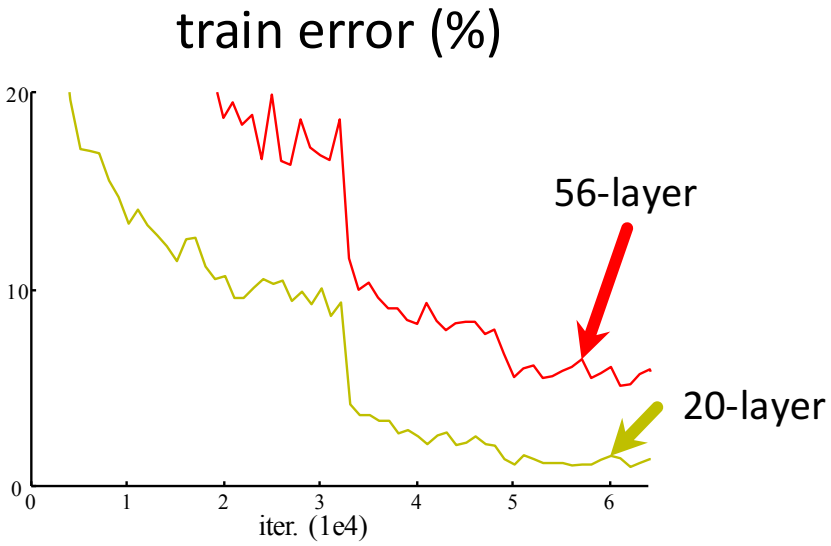


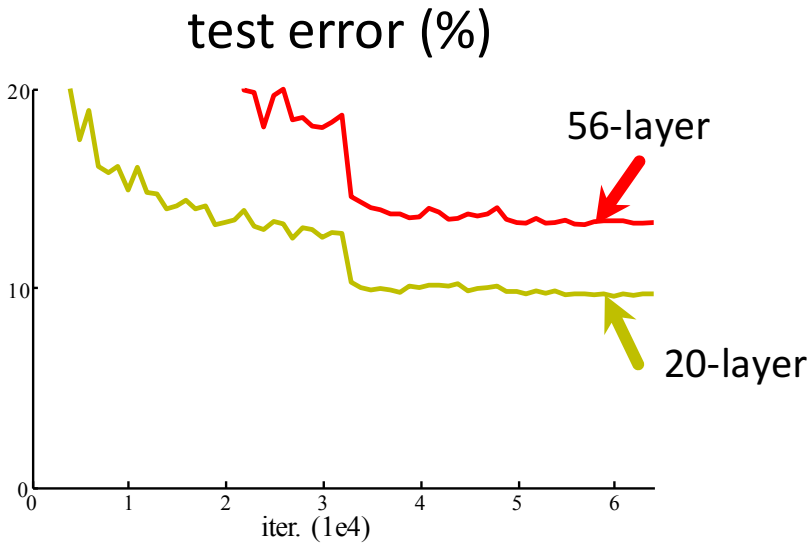


Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

Going even deeper

- Is learning better networks as simple as stacking more layers?



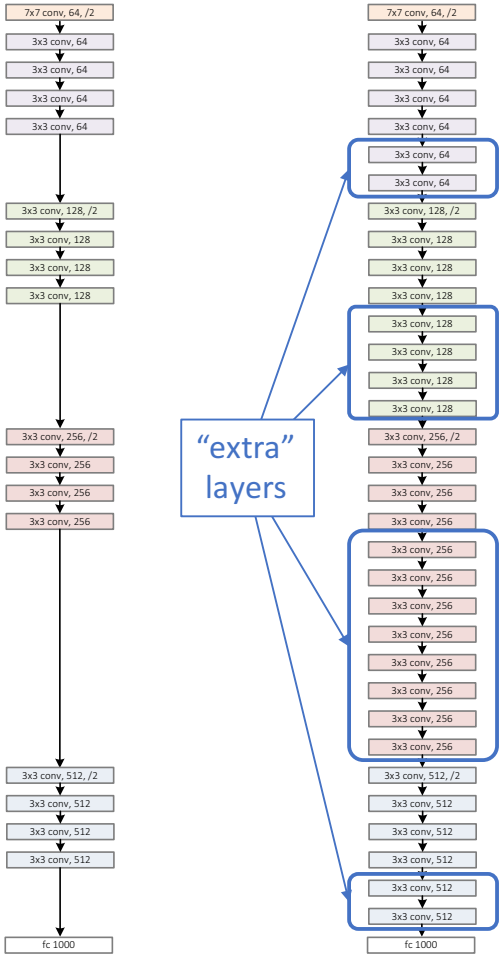


- 56-layer net has **higher training error** and test error than 20-layer net

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

Going even deeper

a shallower model (18 layers)



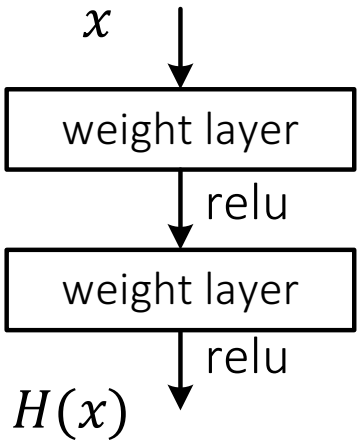
a deeper counterpart (34 layers)

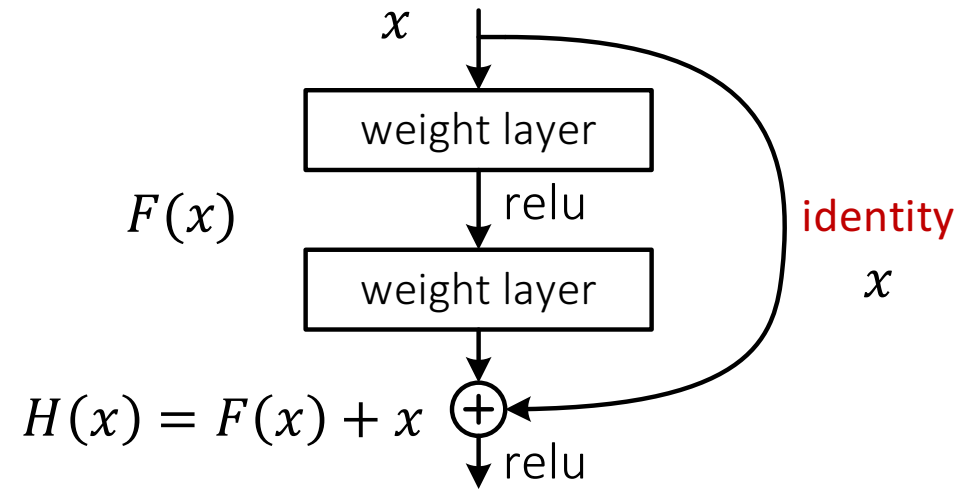
- Richer solution space
- A deeper model should not have **higher training error**
- A solution *by construction*:
  - original layers: copied from a learned shallower model
  - extra layers: set as identity
  - at least the same training error
- Optimization difficulties: solvers cannot find the solution when going deeper…

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

ResNet

- Residual Neural Network (ResNet): Kaiming He et al. CVPR, 2016



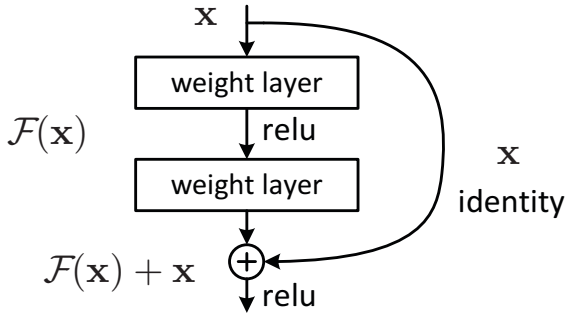


- Winner ILSVRC-2015

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

ResNet

- Residual Neural Network (ResNet): Kaiming He et al. CVPR, 2016

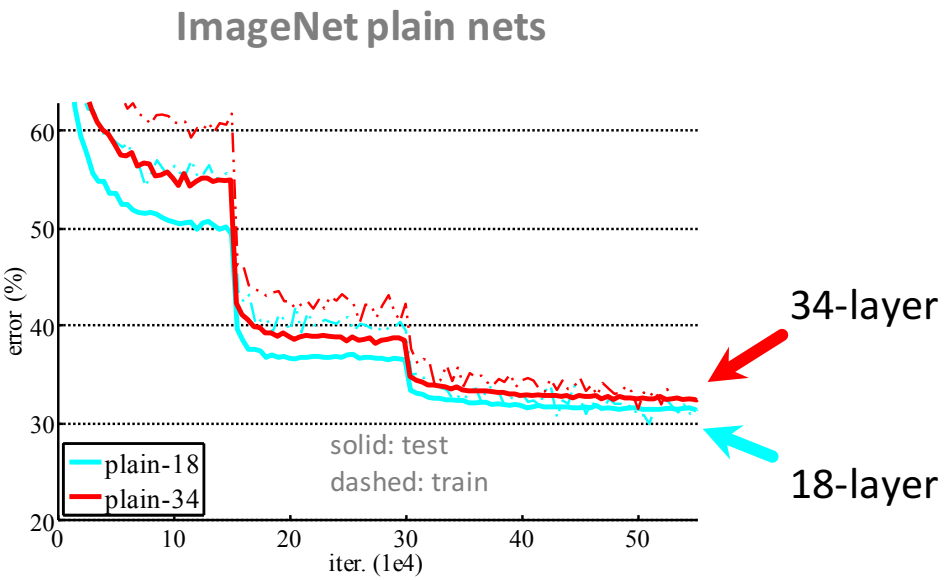


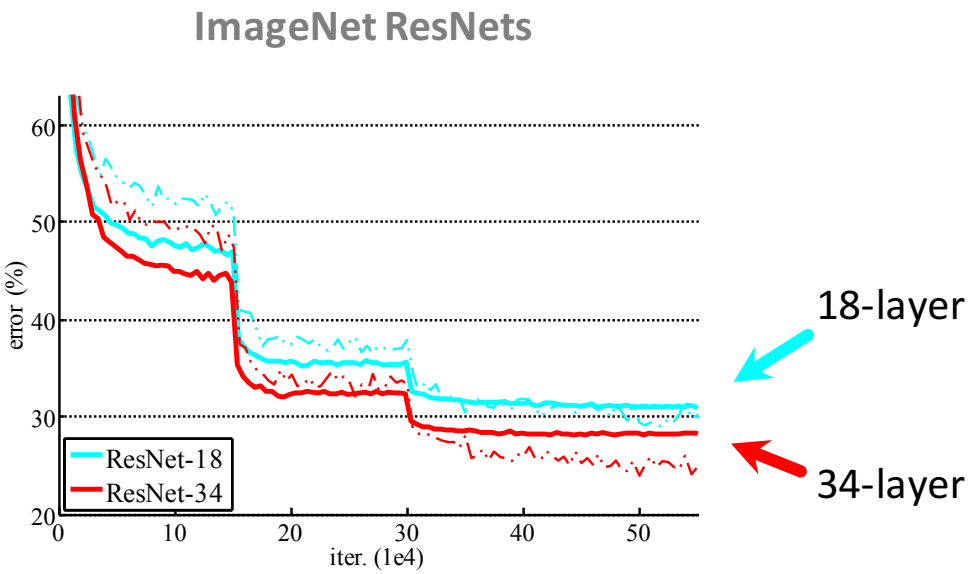
- Standard networks learn $F(x)$
- Instead, residual networks learn

$$H(x) = F(x) - x \Leftrightarrow F(x) = H(x) + x$$

Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

ResNet

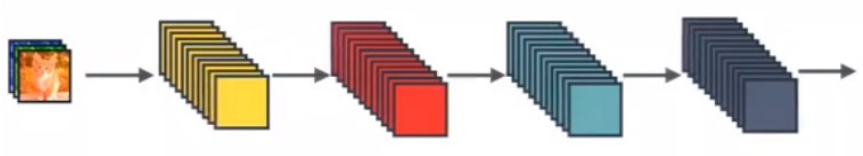




Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

Densenet

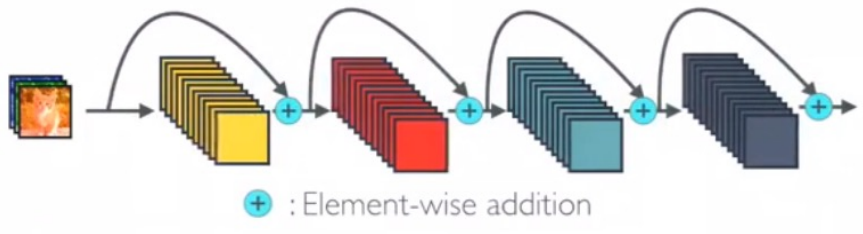
- Densenet: Densely Connected Convolutional Networks (G. Huang et al., CVPR 2017 – Best paper)
- Dense Blocks



Standard ConvNet

Densenet

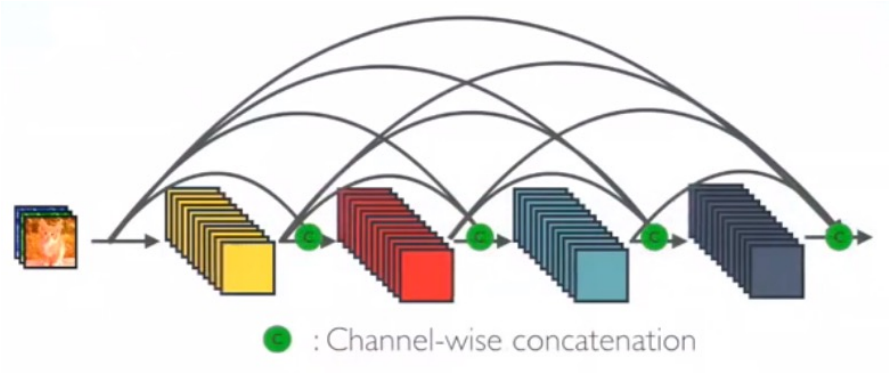
- Densenet: Densely Connected Convolutional Networks (G. Huang et al., CVPR 2017 – Best paper)
- Dense Blocks



ResNet

Densenet

- Densenet: Densely Connected Convolutional Networks (G. Huang et al., CVPR 2017 – Best paper)
- Dense Blocks



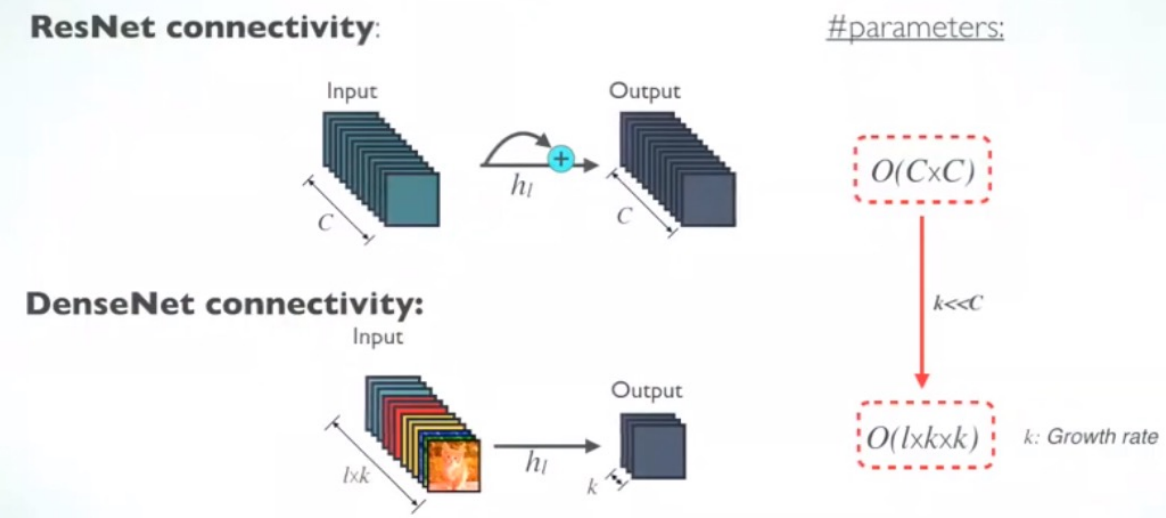
One block in Densenet

Densenet

- Densenet: Densely Connected Convolutional Networks (G. Huang et al., CVPR 2017 – Best paper)
- Dense Blocks

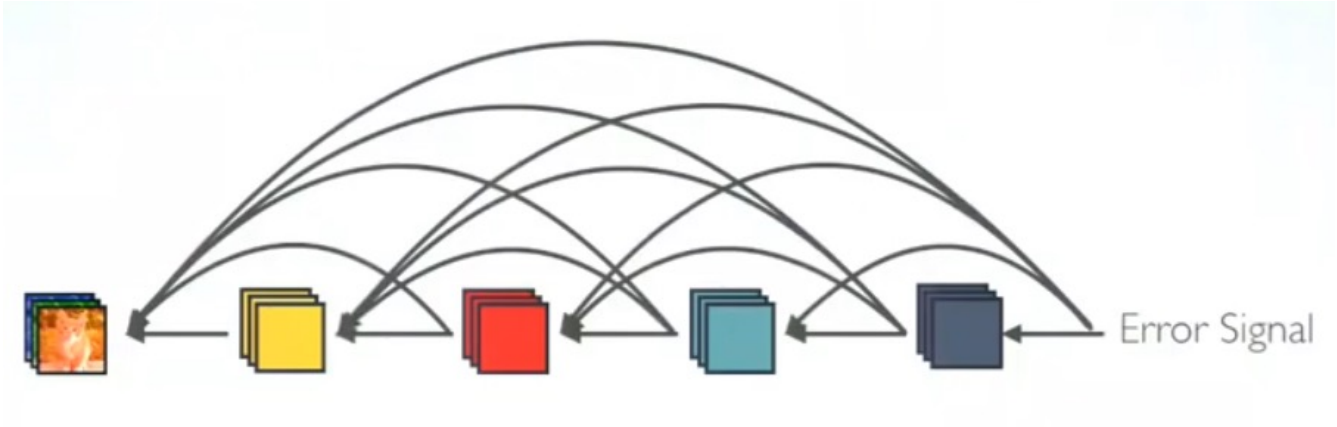
Densenet

- Densenet: Densely Connected Convolutional Networks (G. Huang et al., CVPR 2017 – Best paper)
- Advantages:
  – Parameter & Computational Efficiency



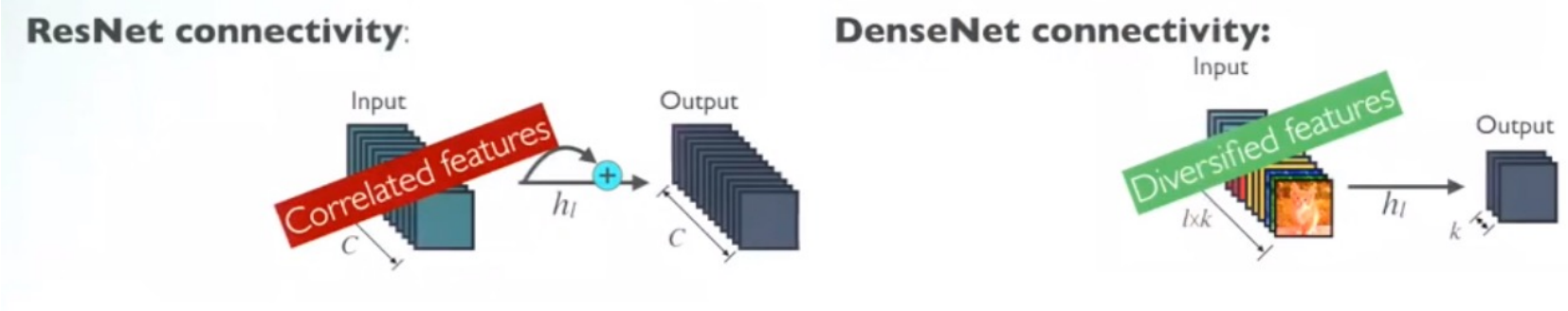
Densenet

- Densenet: Densely Connected Convolutional Networks (G. Huang et al., CVPR 2017 – Best paper)
- Advantages:
  – Implicit deep supervision ➔ strong gradient flow

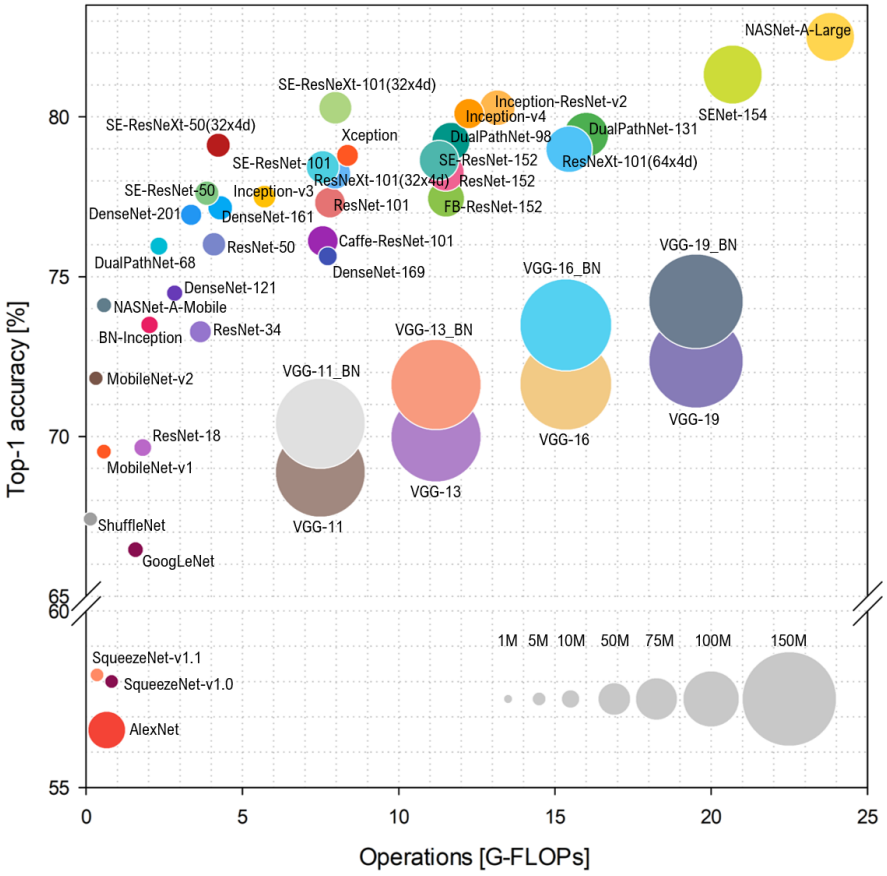


Densenet

- Densenet: Densely Connected Convolutional Networks (G. Huang et al., CVPR 2017 – Best paper)
- Advantages:
  – More diversified features



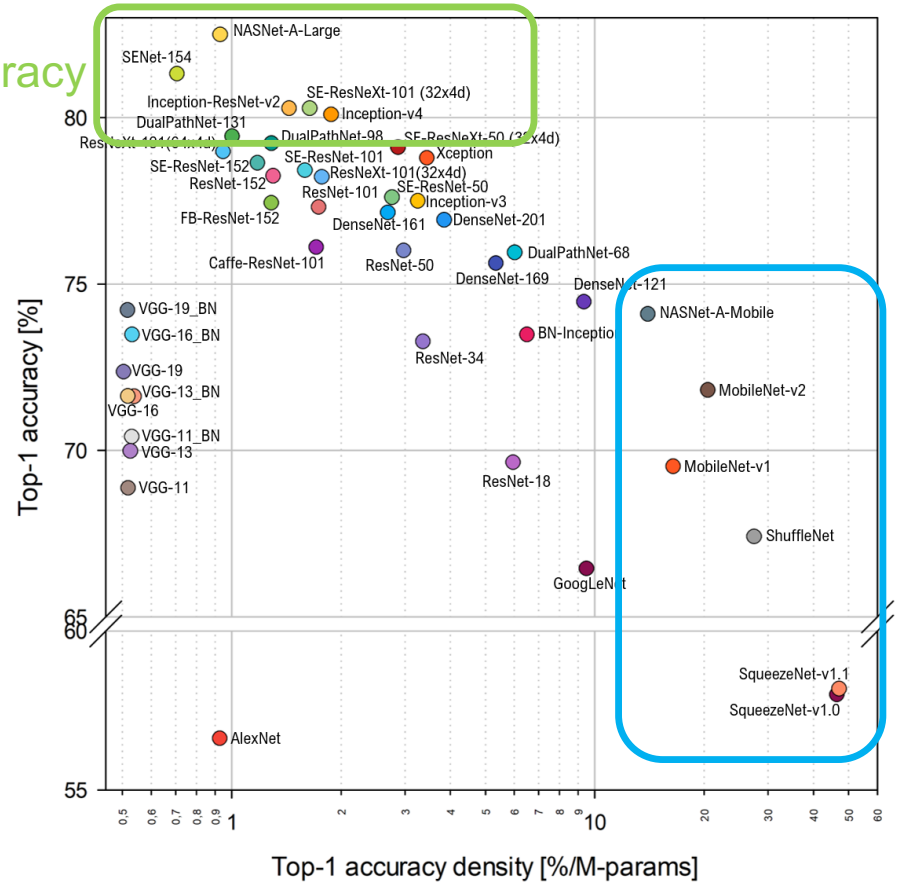
State-of-the-art 2018 – and still going strong



S. Bianco, R. Cadene, L. Celona and P. Napoletano, "Benchmark Analysis of Representative Deep Neural Network Architectures," in IEEE Access, vol. 6, pp. 64270-64277, 2018, doi: 10.1109/ACCESS.2018.2877890.

State-of-the-art 2018 – and still going strong

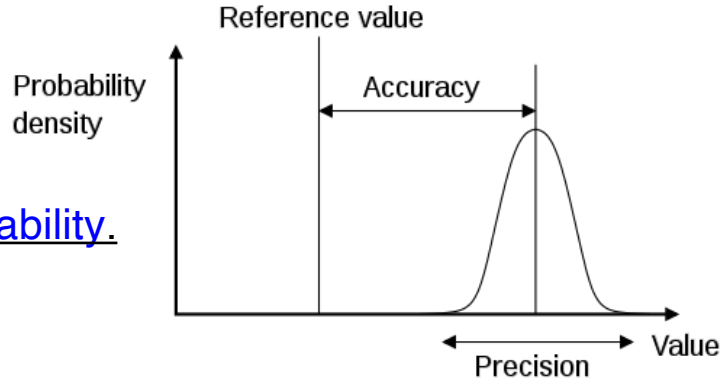
optimized for accuracy



optimized for size

S. Bianco, R. Cadene, L. Celona and P. Napoletano, "Benchmark Analysis of Representative Deep Neural Network Architectures," in IEEE Access, vol. 6, pp. 64270-64277, 2018, doi: 10.1109/ACCESS.2018.2877890.

How to assess classification performance?



- **Precision**
  - is a description of random errors, a measure of statistical variability.
  - the repeatability, or reproducibility of the measurement
- **Accuracy** (two definitions)
  - Description of systematic errors, a measure of statistical bias; as these cause a difference between a result and a "true" value, ISO calls this trueness.
  - Alternatively, ISO defines accuracy as describing a combination of both types of observational error above (random and systematic), so high accuracy requires both high precision and high trueness
- **Robustness**
  - refers to the degradation in performance with respect to varying noise levels or other imaging artefacts

How to assess classification performance: Confusion Matrix

- True positive (TP)
  – eqv. with hit
- True negative (TN)
  – eqv. with correct rejection
- False positive (FP)
  – eqv. with false alarm, Type I error
- False negative (FN)
  – eqv. with miss, Type II error

		Predicted condition	
	Total population = P + N	**Positive (PP)**	**Negative (PN)**
Actual condition	**Positive (P)**	**True positive (TP),** hit	**False negative (FN),** type II error, miss, underestimation
	Negative (N)	**False positive (FP),** type I error, false alarm, overestimation	**True negative (TN),** correct rejection

Accuracy, Precision, Recall, ...

Accuracy

$$ACC = \frac{TP+TN}{P+N},\ P = TP + FN,\ N = TN + FP$$

Precision or **positive predictive value**

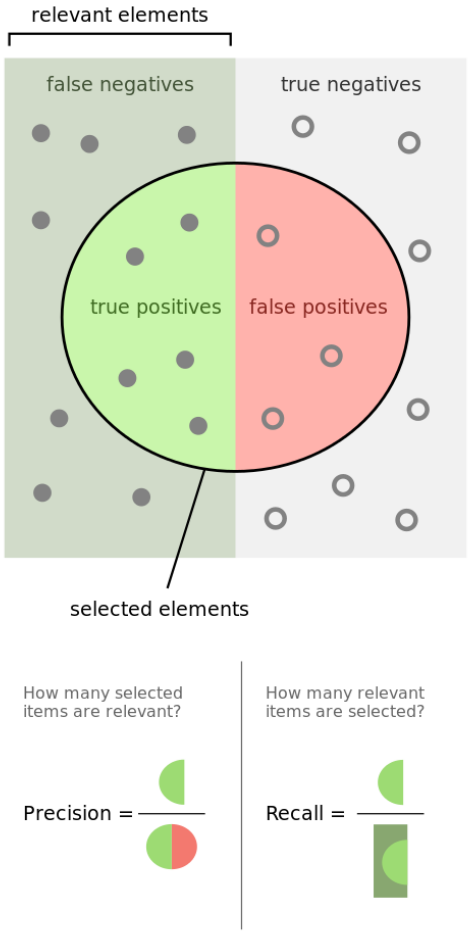
$$PPV = \frac{TP}{TP + FP}$$

Recall, sensitivity, hit rate or **true positive rate**

$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN}$$

Specificity or **true negative rate**

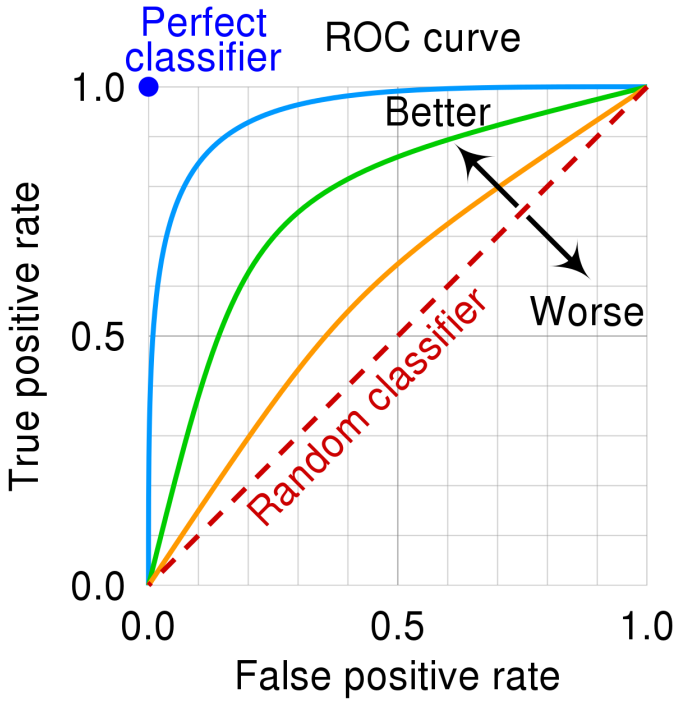
$$TNR = \frac{TN}{N} = \frac{TN}{TN + FP}$$



		Predicted condition			
	Total population $= P + N$	**Positive (PP)**	**Negative (PN)**	Informedness, bookmaker informedness (BM) $= TPR + TNR - 1$	Prevalence threshold (PT) $= \frac{\sqrt{TPR \times FPR} - FPR}{TPR - FPR}$
Actual condition	**Positive (P)**	**True positive (TP),** **hit**	**False negative (FN),** **type II error, miss, underestimation**	True positive rate (TPR), recall, sensitivity (SEN), probability of detection, hit rate, power $= \frac{TP}{P} = 1 - FNR$	False negative rate (FNR), miss rate $= \frac{FN}{P} = 1 - TPR$
	Negative (N)	**False positive (FP),** **type I error, false alarm, overestimation**	**True negative (TN),** **correct rejection**	False positive rate (FPR), probability of false alarm, fall-out $= \frac{FP}{N} = 1 - TNR$	True negative rate (TNR), specificity (SPC), selectivity $= \frac{TN}{N} = 1 - FPR$
	Prevalence $= \frac{P}{P + N}$	Positive predictive value (PPV), precision $= \frac{TP}{PP} = 1 - FDR$	False omission rate (FOR) $= \frac{FN}{PN} = 1 - NPV$	Positive likelihood ratio (LR+) $= \frac{TPR}{FPR}$	Negative likelihood ratio (LR−) $= \frac{FNR}{TNR}$
	Accuracy (ACC) $= \frac{TP + TN}{P + N}$	False discovery rate (FDR) $= \frac{FP}{PP} = 1 - PPV$	Negative predictive value (NPV) $= \frac{TN}{PN}$ $= 1 - FOR$	Markedness (MK), deltaP (Δp) $= PPV + NPV - 1$	Diagnostic odds ratio (DOR) $= \frac{LR+}{LR-}$
	Balanced accuracy (BA) $= \frac{TPR + TNR}{2}$	F_1 score $= \frac{2PPV \times TPR}{PPV + TPR} = \frac{2TP}{2TP + FP + FN}$	Fowlkes–Mallows index (FM) $= \sqrt{PPV \times TPR}$	Matthews correlation coefficient (MCC) $= \sqrt{TPR \times TNR \times PPV \times NPV} - \sqrt{FNR \times FPR \times FOR \times FDR}$	Threat score (TS), critical success index (CSI), Jaccard index $= \frac{TP}{TP + FN + FP}$

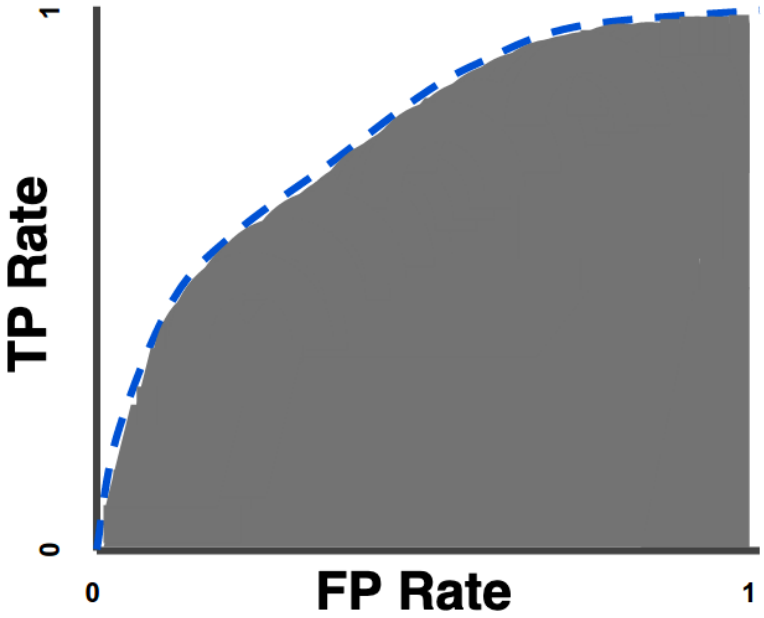
ROC – Receiver Operator Characteristic Curve

A			B			C			C′		
TP=63	FN=37	100	TP=77	FN=23	100	TP=24	FN=76	100	TP=76	FN=24	100
FP=28	TN=72	100	FP=77	TN=23	100	FP=88	TN=12	100	FP=12	TN=88	100
91	109	200	154	46	200	112	88	200	88	112	200
TPR = 0.63			TPR = 0.77			TPR = 0.24			TPR = 0.76		
FPR = 0.28			FPR = 0.77			FPR = 0.88			FPR = 0.12		
PPV = 0.69			PPV = 0.50			PPV = 0.21			PPV = 0.86		
F1 = 0.66			F1 = 0.61			F1 = 0.23			F1 = 0.81		
ACC = 0.68			ACC = 0.50			ACC = 0.18			ACC = 0.82		



AUC – Area under the ROC Curve

- AUC ranges in value from 0 to 1:
  - A model whose predictions are 100% wrong has an AUC of 0.0
  - A model whose predictions are 100% correct has an AUC of 1.0.
- AUC is **scale-invariant**. It measures how well predictions are ranked, rather than their absolute values.
- AUC is **classification-threshold-invariant**. It measures the quality of the model's predictions irrespective of what classification threshold is chosen.



Imperial College London

Case study: Classification of x-ray images

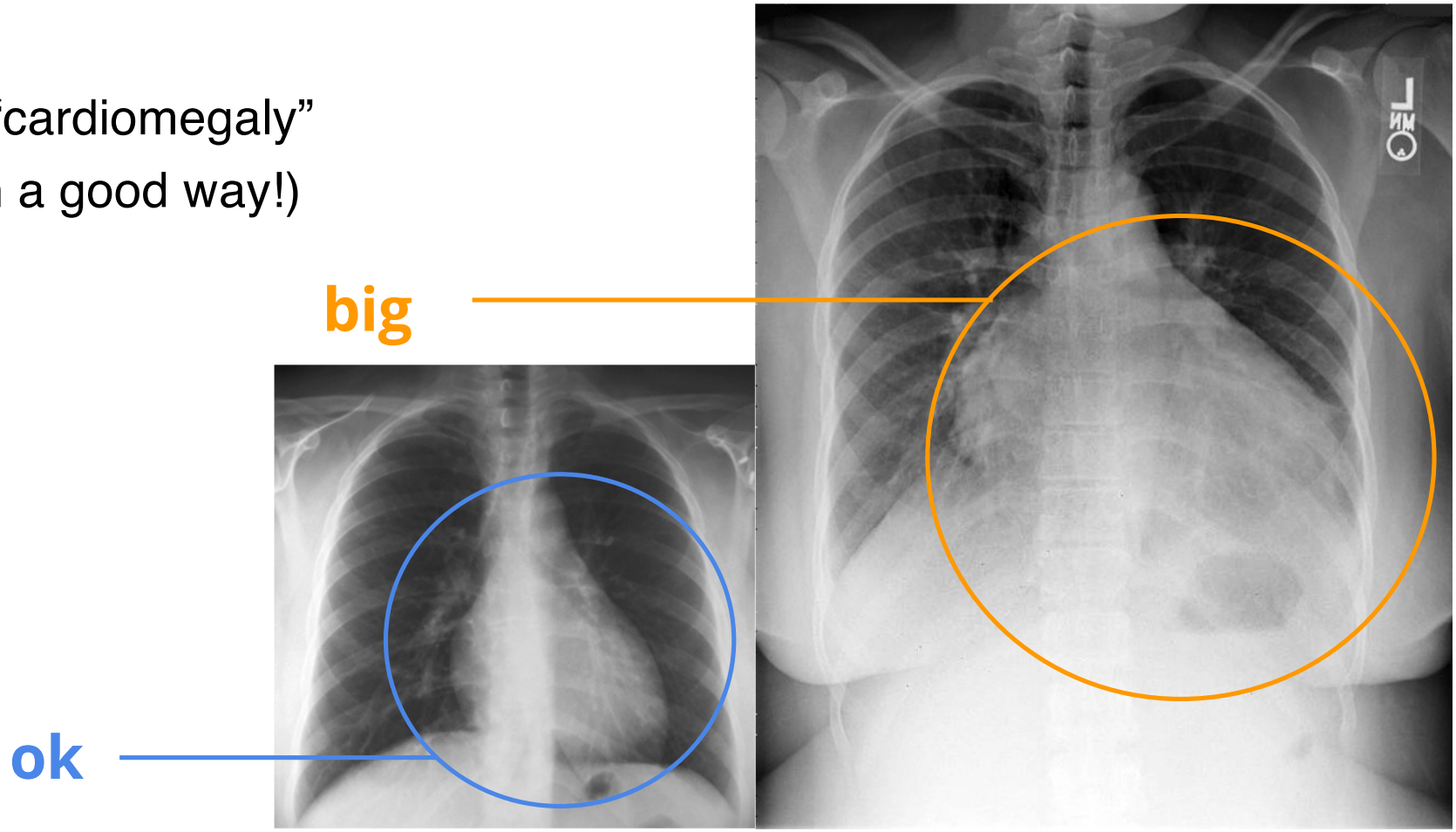
Some slides from Alistair Johnson, Hospital for Sick Children in Toronto, Canada

Chest radiographs

- X-rays
  - Visualizes the lungs
  - Visualizes the heart
- What do radiologists look for in these x-rays?

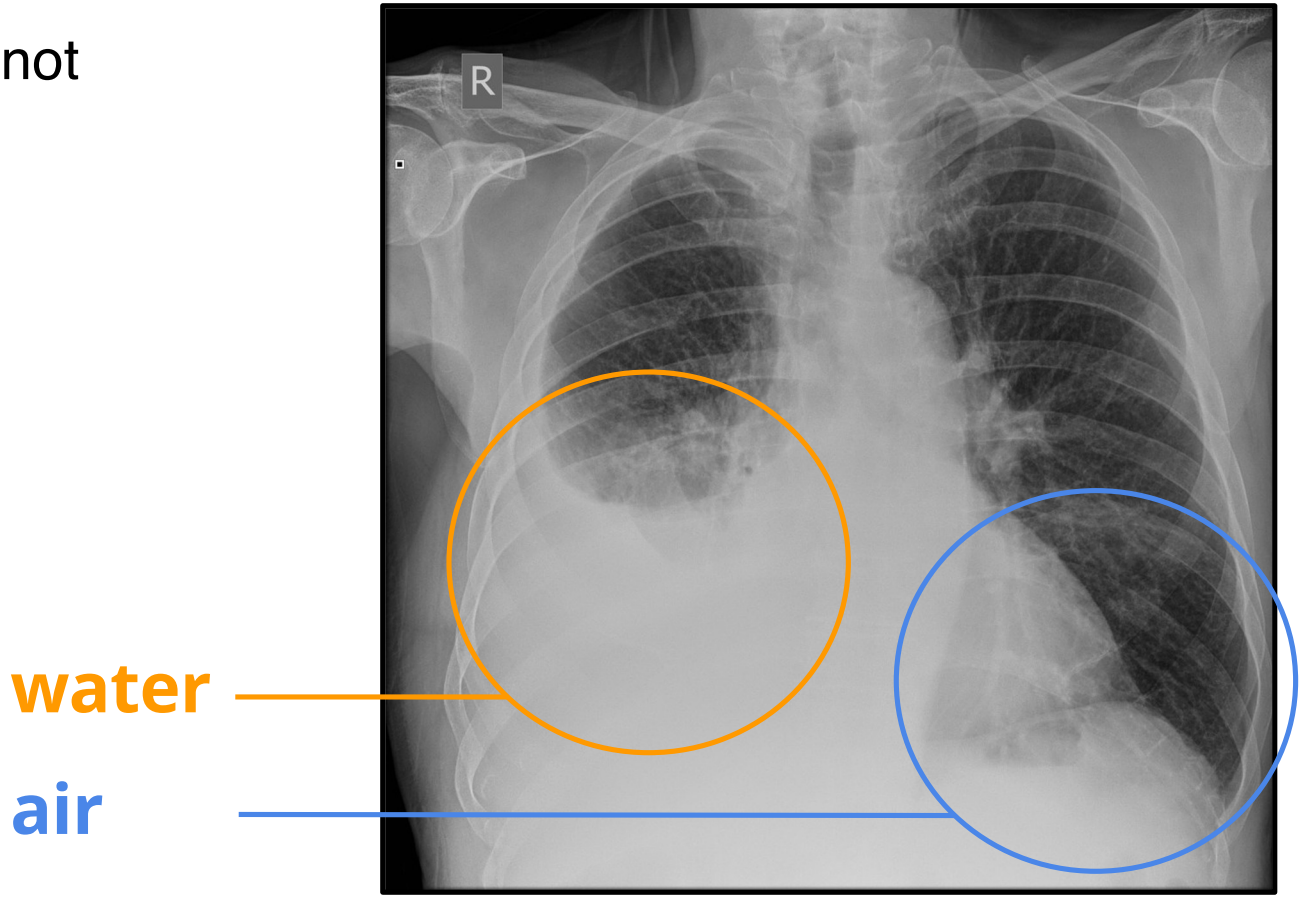
What do we look for in a CXR?

- big heart, “cardiomegaly”
  (and not in a good way!)



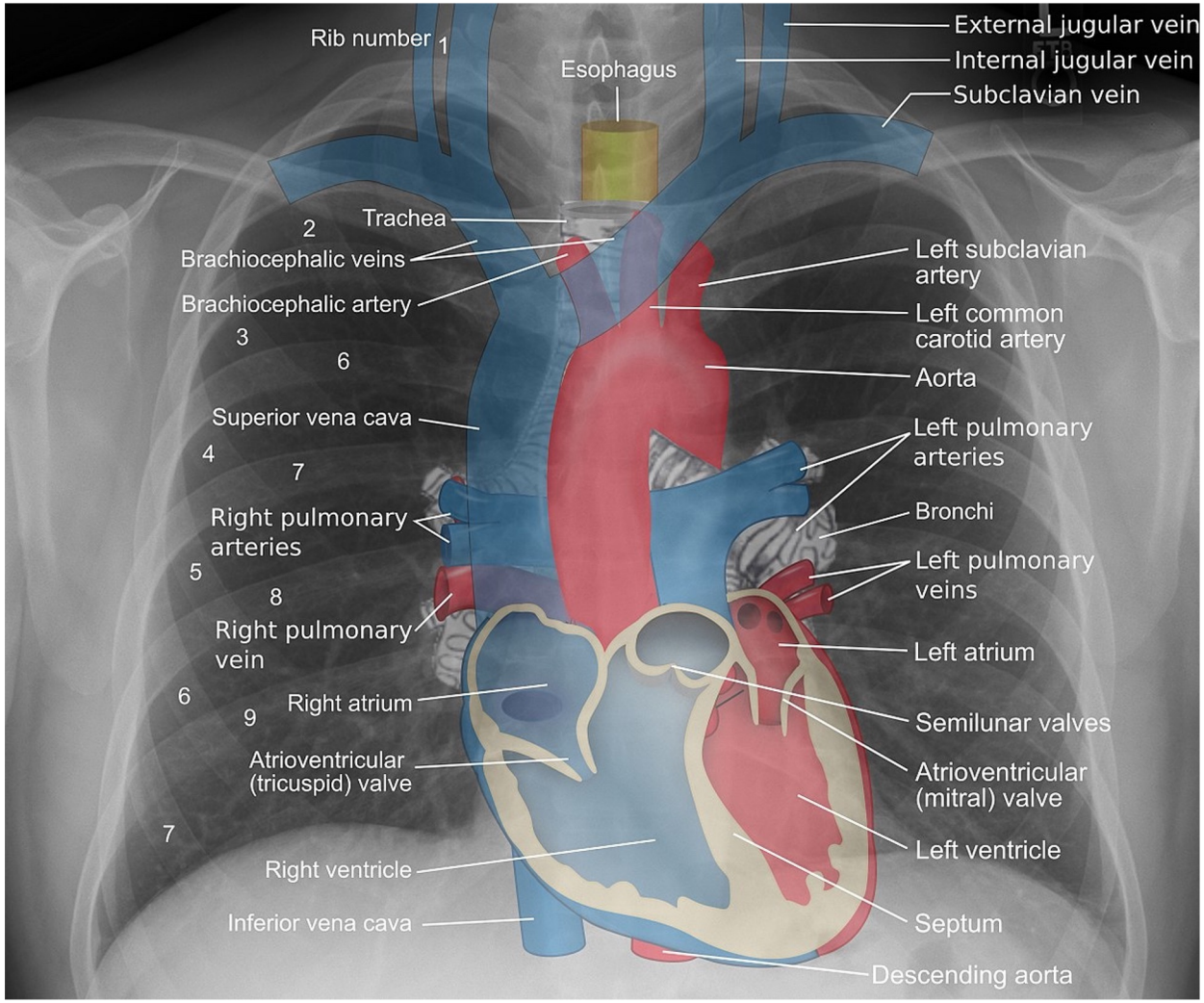
What do we look for in a chest x-ray?

- Water where it is not supposed to be



Chest radiographs

Radiologists have a strong mental model of what a CXR **should look like**



M. Häggström A pictorial essay: Radiology of lines and tubes in the intensive care unit". Indian Journal of Radiology and Imaging 21(3):182.

NIH x-ray chest dataset

- 30,000+ patients
- 100,000+ images
- Each associated with 14 labels
  - Derived automatically from the free-text report
- Freely, publicly available

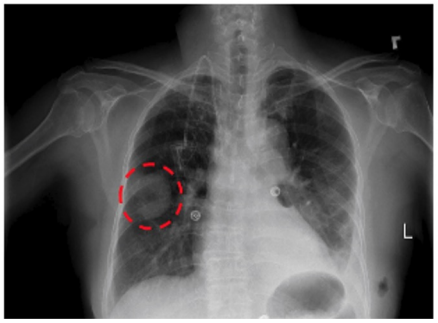
Media Advisory Wednesday, September 27, 2017

NIH Clinical Center provides one of the largest publicly available chest x-ray datasets to scientific community

The dataset of scans is from more than 30,000 patients, including many with advanced lung disease.

What

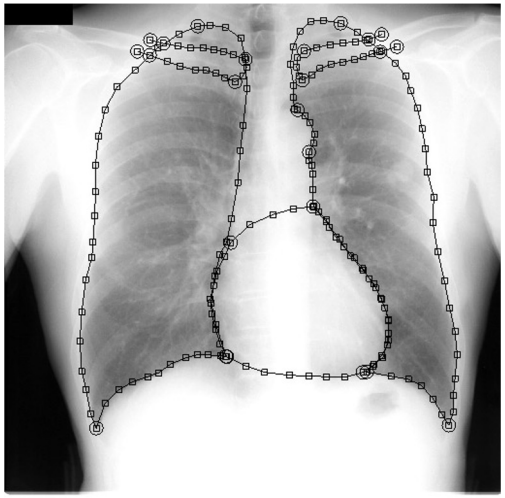
The NIH Clinical Center recently released over 100,000 anonymized chest x-ray images and their corresponding data to the scientific community. The release will allow researchers across the country and around the world to freely access the datasets and increase their ability to teach computers how to detect and diagnose disease. Ultimately, this artificial intelligence mechanism can lead to clinicians making better diagnostic decisions for patients.

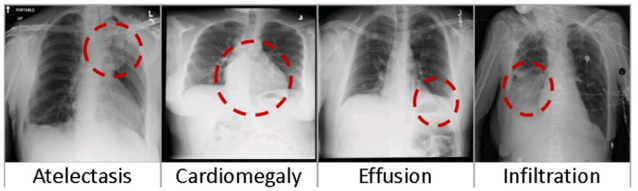


https://www.nih.gov/news-events/news-releases/nih-clinical-center-provides-one-largest-publicly-available-chest-x-ray-datasets-scientific-community

Others datasets

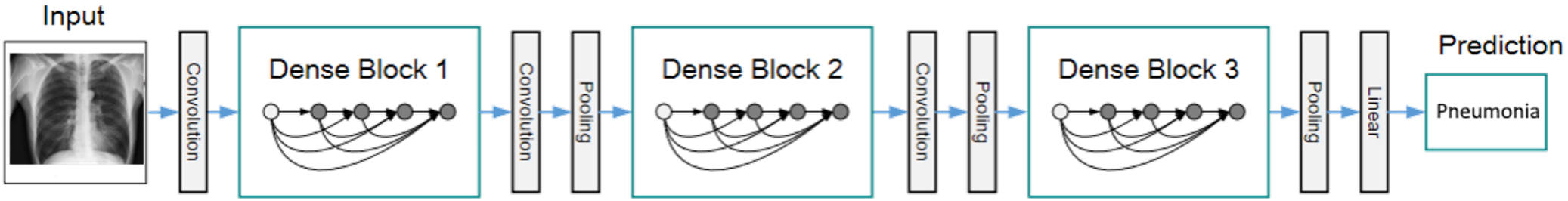
- Currently available chest x-ray datasets
  - JSRT Database - 247 DICOMs, have heart + lung segmentation
  - Open-I Indiana University CXR - 8121 DICOMs, 3996 reports
  - ChestXray14 (ChestXray8) - 112,120 PNGs, 14 labels
  - MIMIC-CXR-JPG - 369,188 JPGs, 14 labels
  - MIMIC-CXR - same as above, full resolution DICOMs + actual text
  - VinDr-CXR - 18,000 PA views, 28 labels (22 labels are BBs)
  - CheXpert - 224,316 JPGs, 14 labels
  - PadChest - 160,000 PNGs
    - 174 findings / 19 dx / 104 anatomy
    - UMLS coded
  - RSNA + Kaggle - too many to list!





CheXNet – Classification of chest pathologies

- Classify chest x-rays using a large labeled dataset
- CNN-based approach: DenseNet-121
- Compare to domain experts (radiologists)



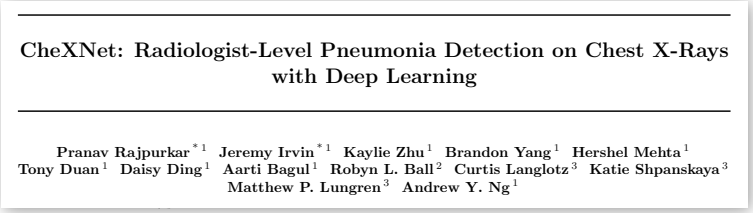
CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning

Pranav Rajpurkar [*1] Jeremy Irvin [*1] Kaylie Zhu [1] Brandon Yang [1] Hershel Mehta [1]
Tony Duan [1] Daisy Ding [1] Aarti Bagul [1] Robyn L. Ball [2] Curtis Langlotz [3] Katie Shpanskaya [3]
Matthew P. Lungren [3] Andrew Y. Ng [1]

CheXNet – Classification of chest pathologies

	F1 Score (95% CI)
Radiologist 1	0.383 (0.309, 0.453)
Radiologist 2	0.356 (0.282, 0.428)
Radiologist 3	0.365 (0.291, 0.435)
Radiologist 4	0.442 (0.390, 0.492)
Radiologist Avg.	0.387 (0.330, 0.442)
CheXNet	0.435 (0.387, 0.481)

Recall that the F1 metric is the harmonic average of precision and recall



CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning

Pranav Rajpurkar [*1] Jeremy Irvin [*1] Kaylie Zhu [1] Brandon Yang [1] Hershel Mehta [1]
Tony Duan [1] Daisy Ding [1] Aarti Bagul [1] Robyn L. Ball [2] Curtis Langlotz [3] Katie Shpanskaya [3]
Matthew P. Lungren [3] Andrew Y. Ng [1]

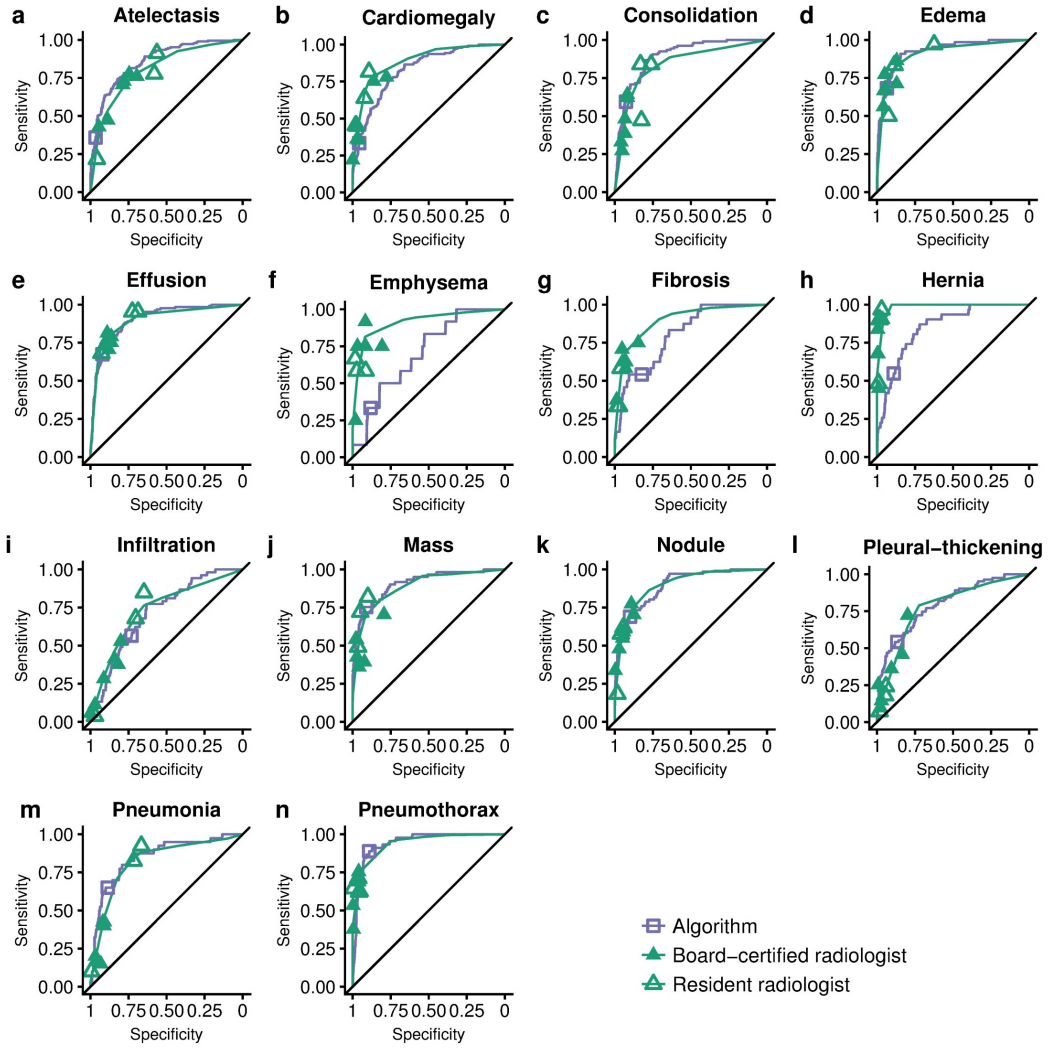
CheXNet – Classification of chest pathologies

- ChexNet outputs a vector of binary labels indicating the absence or presence of each of the following 14 pathology classes:
  - Atelectasis, Cardiomegaly, Consolidation, Edema, Effusion, Emphysema, Fibrosis, Hernia, Infiltration, Mass, Nodule, Pleural Thickening, Pneumonia, and Pneumothorax.

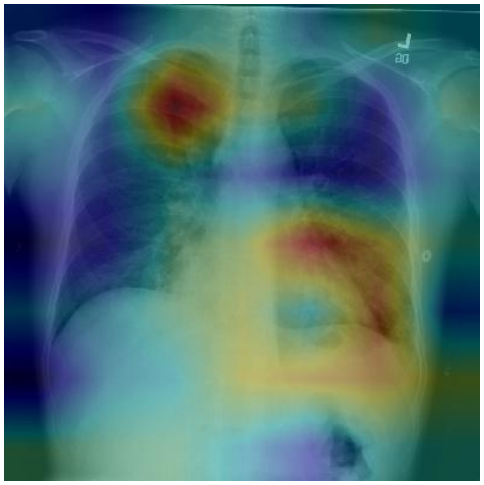
$$L(X, y) = \sum_{c=1}^{14} [-y_c \log p(Y_c = 1|X) - (1 - y_c) \log p(Y_c = 0|X)],$$

Pathology	Wang et al. (2017)	Yao et al. (2017)	CheXNet (ours)
Atelectasis	0.716	0.772	**0.8094**
Cardiomegaly	0.807	0.904	**0.9248**
Effusion	0.784	0.859	**0.8638**
Infiltration	0.609	0.695	**0.7345**
Mass	0.706	0.792	**0.8676**
Nodule	0.671	0.717	**0.7802**
Pneumonia	0.633	0.713	**0.7680**
Pneumothorax	0.806	0.841	**0.8887**
Consolidation	0.708	0.788	**0.7901**
Edema	0.835	0.882	**0.8878**
Emphysema	0.815	0.829	**0.9371**
Fibrosis	0.769	0.767	**0.8047**
Pleural Thickening	0.708	0.765	**0.8062**
Hernia	0.767	0.914	**0.9164**

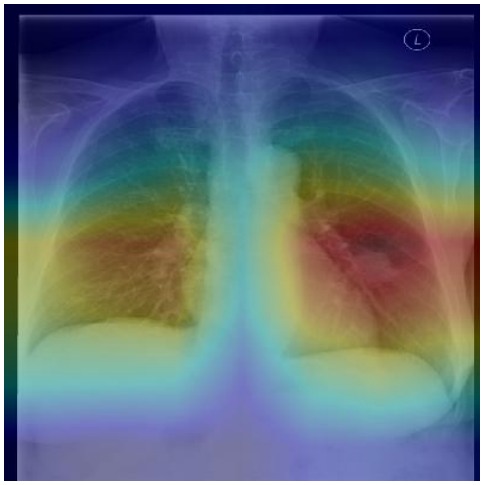
CheXNet – Classification of chest pathologies



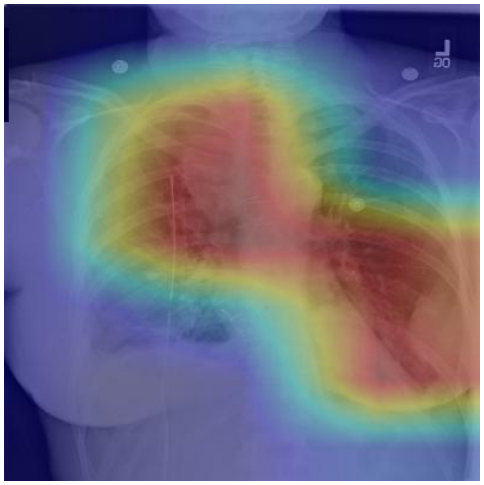
CheXNet – Classification of chest pathologies



(a) Patient with multifocal community acquired pneumonia. The model correctly detects the airspace disease in the left lower and right upper lobes to arrive at the pneumonia diagnosis.



(b) Patient with a left lung nodule. The model identifies the left lower lobe lung nodule and correctly classifies the pathology.

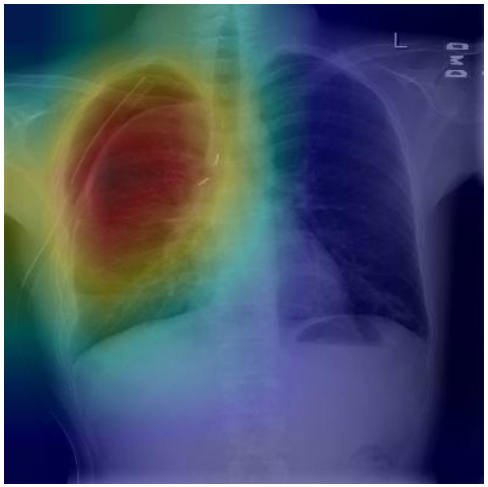


(c) Patient with primary lung malignancy and two large masses, one in the left lower lobe and one in the right upper lobe adjacent to the mediastinum. The model correctly identifies both masses in the X-ray.

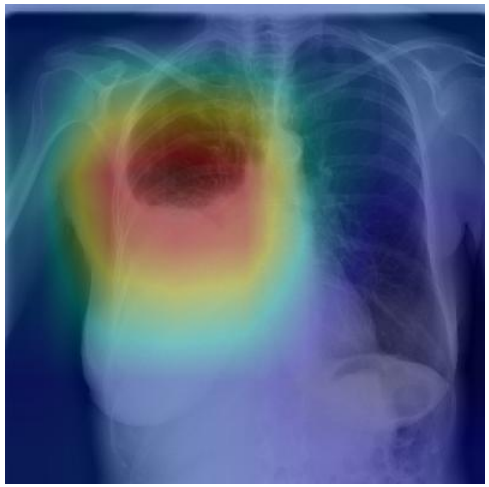
CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning

Pranav Rajpurkar [*1] Jeremy Irvin [*1] Kaylie Zhu [1] Brandon Yang [1] Hershel Mehta [1]
Tony Duan [1] Daisy Ding [1] Aarti Bagul [1] Robyn L. Ball [2] Curtis Langlotz [3] Katie Shpanskaya [3]
Matthew P. Lungren [3] Andrew Y. Ng [1]

CheXNet – Classification of chest pathologies



(d) Patient with a right-sided pneumothroax and chest tube. The model detects the abnormal lung to correctly predict the presence of pneumothorax (collapsed lung).



(e) Patient with a large right pleural effusion (fluid in the pleural space). The model correctly labels the effusion and focuses on the right lower chest.

(f) Patient with congestive heart failure and cardiomegaly (enlarged heart). The model correctly identifies the enlarged cardiac silhouette.

CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning

Pranav Rajpurkar [*1] Jeremy Irvin [*1] Kaylie Zhu [1] Brandon Yang [1] Hershel Mehta [1]
Tony Duan [1] Daisy Ding [1] Aarti Bagul [1] Robyn L. Ball [2] Curtis Langlotz [3] Katie Shpanskaya [3]
Matthew P. Lungren [3] Andrew Y. Ng [1]

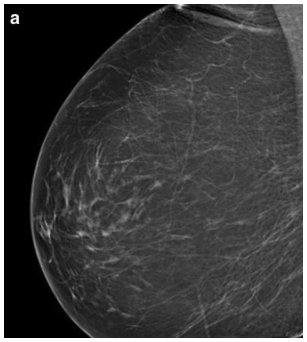
Mammography screening: Background

X-ray imaging of the human breast

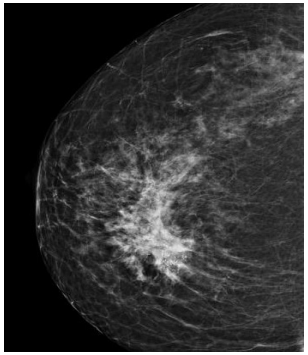
- Goal:
  – Early detection of breast cancer
- Procedure:
  – Imaging of each breast separately, 2 images from different angles per breast
- In Germany (and USA) mammography screening every two years recommended for all women between the age of 50 and 74; earlier in case of relatives with breast cancer
- Radiological evaluation of
  – Breast density
  – Presence of focal masses and malignancy risk estimation of those lesions

Radiological classification task: Risk evaluation

- BI-RADS – Breast Imaging-Reporting and Data System
- Five relevant classes corresponding to certain risk of malignancy
  - 1 – lowest risk
  - 5 – highest risk
- Radiological criteria: shape, density and margin of lesion, presence and distribution of calcifications, architectural distortion, asymmetry, intramammarial lymph nodes, skin retraction
- Applied in mammography images, breast MRI and ultrasound images



BI-RADS I[1]

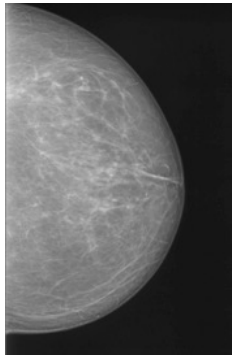


BI-RADS V[2]

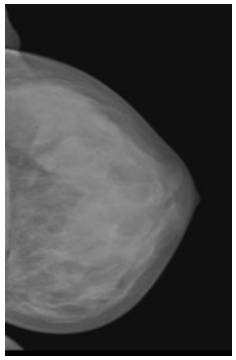
[1]Moghbel, Mehrdad & Ooi, Chia-Yee & Ismail, Nordinah & Hau, Y.W. & Memari, Nogol. (2020). A review of breast boundary and pectoral muscle segmentation methods in computer-aided detection/diagnosis of breast mammography. Artificial Intelligence Review. 53. 10.1007/s10462-019-09721-8.

Radiological classification task: Breast density / ACR

- ACR – American Collage of Radiology, classification for breast density
- Breast density refers to ratio of glandular tissue in breast relative to fat
  - – Dense breasts as risk factor for breast cancer
  - – Sensitivity for detecting focal lesions higher in fatty tissue, thus validity of mammography lower in women with more dense breasts
    → Ultrasound screening or breast MRI might be more beneficial
- Four classes
  - – a – almost entirely fatty
  - – d – extremely dense breast
- Used only in mammography images, but similar classification system exists for breast MRI



ACR MG-a[1]

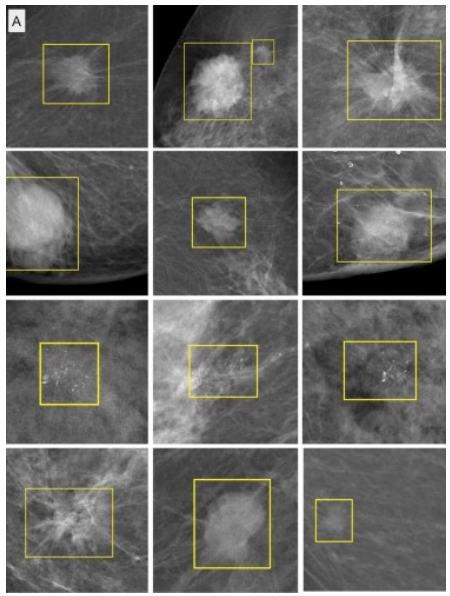


ACR MG-d[1]

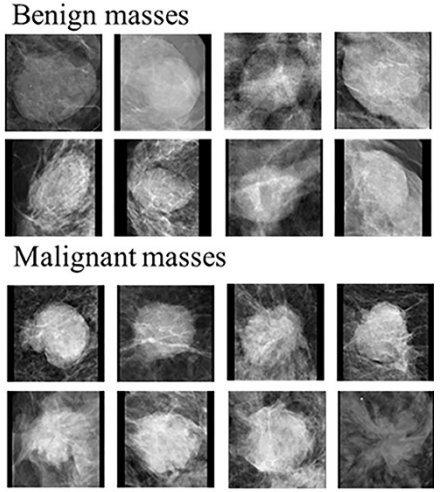
[1]Wengert, Georg & Helbich, Thomas & Leithner, Doris & Morris, Elizabeth & Baltzer, Pascal & Pinker, Katja. (2019). Multimodality Imaging of Breast Parenchymal Density and Correlation with Risk Assessment. Current Breast Cancer Reports.

Mammography screening: Possible AI applications

- Lesion detection



- Classification of lesions
- Malignant vs benign
- Multi-class classification (BI-RADS, ARC`B



Ribli, D., Horváth, A., Unger, Z. *et al.* Detecting and classifying lesions in mammograms with Deep Learning. *Sci Rep* **8,** 4165 (2018).
Cui Y, Li Y, Xing D, Bai T, Dong J and Zhu J (2021) Improving the Prediction of Benign or Malignant Breast Masses Using a Combination of Image Biomarkers and Clinical Parameters. *Front. Oncol.* 11:

Mammography screening: Current status of CAD

- First implementation of computer aided diagnosis algorithms for mammography images in the late 1990's
- Mostly failed in clinical usage due to some major limitations:
  - No significant increase in radiologists' performance
  - High rate of false positives
  - Poor integration into existing clinical workflows
- Deep learning has shown erformance equal or superior to human readers achieved by several groups
- Challenges still to be overcome:
  - Poor generalizability of algorithms outside studies
  - Integration into clinical workflows
  - Interpretability optimization

LETTER

doi:10.1038/nature21056

Dermatologist-level classification of skin cancer with deep neural networks

Andre Esteva[1*], Brett Kuprel[1*], Roberto A. Novoa[2,3], Justin Ko[2], Susan M. Swetter[2,4], Helen M. Blau[5] & Sebastian Thrun[6]

Skin cancer, the most common human malignancy[1–3], is primarily diagnosed visually, beginning with an initial clinical screening and followed potentially by dermoscopic analysis, a biopsy and histopathological examination. Automated classification of skin lesions using images is a challenging task owing to the fine-grained variability in the appearance of skin lesions. Deep convolutional neural networks (CNNs)[4,5] show potential for general and highly variable tasks across many fine-grained object categories[6–11]. Here we demonstrate classification of skin lesions using a single CNN, trained end-to-end from images directly, using only pixels and disease labels as inputs. We train a CNN using a dataset of 129,450 clinical images—two orders of magnitude larger than previous datasets[12]—consisting of 2,032 different diseases. We test its performance against 21 board-certified dermatologists on biopsy-proven clinical images with two critical binary classification use cases: keratinocyte carcinomas versus benign seborrheic keratoses; and malignant melanomas versus benign nevi. The first case represents the identification of the most common cancers, the second represents the identification of the deadliest skin cancer. The CNN achieves performance on par with all tested experts across both tasks, demonstrating an artificial intelligence capable of classifying skin cancer with a level of competence comparable to dermatologists. Outfitted with deep neural networks, mobile devices can potentially extend the reach of dermatologists outside of the clinic. It is projected that 6.3 billion smartphone subscriptions will exist by the year 2021 (ref. 13) and can therefore potentially provide low-cost universal access to vital diagnostic care.

There are 5.4 million new cases of skin cancer in the United States[2] every year. One in five Americans will be diagnosed with a cutaneous malignancy in their lifetime. Although melanomas represent fewer than 5% of all skin cancers in the United States, they account for approximately 75% of all skin-cancer-related deaths, and are responsible for over 10,000 deaths annually in the United States alone. Early detection is critical, as the estimated 5-year survival rate for melanoma drops from over 99% if detected in its earliest stages to about 14% if detected in its latest stages. We developed a computational method which may allow medical practitioners and patients to proactively track skin lesions and detect cancer earlier. By creating a novel disease taxonomy, and a disease-partitioning algorithm that maps individual diseases into training classes, we are able to build a deep learning system for automated dermatology.

Previous work in dermatological computer-aided classification[12,14,15] has lacked the generalization capability of medical practitioners owing to insufficient data and a focus on standardized tasks such as dermoscopy[16–18] and histological image classification[19–22]. Dermoscopy images are acquired via a specialized instrument and histological images are acquired via invasive biopsy and microscopy; whereby both modalities yield highly standardized images. Photographic images (for example, smartphone images) exhibit variability in factors such as zoom, angle and lighting, making classification substantially more challenging[23,24]. We overcome this challenge by using a data-driven approach—1.41 million pre-training and training images make classification robust to photographic variability. Many previous techniques require extensive preprocessing, lesion segmentation and extraction of domain-specific visual features before classification. By contrast, our system requires no hand-crafted features; it is trained end-to-end directly from image labels and raw pixels, with a single network for both photographic and dermoscopic images. The existing body of work uses small datasets of typically less than a thousand images of skin lesions[16,18,19], which, as a result, do not generalize well to new images. We demonstrate generalizable classification with a new dermatologist-labelled dataset of 129,450 clinical images, including 3,374 dermoscopy images.

Deep learning algorithms, powered by advances in computation and very large datasets[25], have recently been shown to exceed human performance in visual tasks such as playing Atari games[26], strategic board games like Go[27] and object recognition[6]. In this paper we outline the development of a CNN that matches the performance of dermatologists at three key diagnostic tasks: melanoma classification, melanoma classification using dermoscopy and carcinoma classification. We restrict the comparisons to image-based classification.

We utilize a GoogleNet Inception v3 CNN architecture[9] that was pre-trained on approximately 1.28 million images (1,000 object categories) from the 2014 ImageNet Large Scale Visual Recognition Challenge[6], and train it on our dataset using transfer learning[28]. Figure 1 shows the working system. The CNN is trained using 757 disease classes. Our dataset is composed of dermatologist-labelled images organized in a tree-structured taxonomy of 2,032 diseases, in which the individual diseases form the leaf nodes. The images come from 18 different clinician-curated, open-access online repositories, as well as from clinical data from Stanford University Medical Center. Figure 2a shows a subset of the full taxonomy, which has been organized clinically and visually by medical experts. We split our dataset into 127,463 training and validation images and 1,942 biopsy-labelled test images.

To take advantage of fine-grained information contained within the taxonomy structure, we develop an algorithm (Extended Data Table 1) to partition diseases into fine-grained training classes (for example, amelanotic melanoma and acrolentiginous melanoma). During inference, the CNN outputs a probability distribution over these fine classes. To recover the probabilities for coarser-level classes of interest (for example, melanoma) we sum the probabilities of their descendants (see Methods and Extended Data Fig. 1 for more details).

We validate the effectiveness of the algorithm in two ways, using nine-fold cross-validation. First, we validate the algorithm using a three-class disease partition—the first-level nodes of the taxonomy, which represent benign lesions, malignant lesions and non-neoplastic

[1]Department of Electrical Engineering, Stanford University, Stanford, California, USA. [2]Department of Dermatology, Stanford University, Stanford, California, USA. [3]Department of Pathology, Stanford University, Stanford, California, USA. [4]Dermatology Service, Veterans Affairs Palo Alto Health Care System, Palo Alto, California, USA. [5]Baxter Laboratory for Stem Cell Biology, Department of Microbiology and Immunology, Institute for Stem Cell Biology and Regenerative Medicine, Stanford University, Stanford, California, USA. [6]Department of Computer Science, Stanford University, Stanford, California, USA.
*These authors contributed equally to this work.

PLOS | MEDICINE

RESEARCH ARTICLE

Deep-learning-assisted diagnosis for knee magnetic resonance imaging: Development and retrospective validation of MRNet

Nicholas Bien[1☯*], Pranav Rajpurkar[1☯], Robyn L. Ball[2], Jeremy Irvin[1], Allison Park[1], Erik Jones[1], Michael Bereket[1], Bhavik N. Patel[3], Kristen W. Yeom[3], Katie Shpanskaya[3], Safwan Halabi[3], Evan Zucker[3], Gary Fanton[4], Derek F. Amanatullah[4], Christopher F. Beaulieu[3], Geoffrey M. Riley[3], Russell J. Stewart[3], Francis G. Blankenberg[3], David B. Larson[3], Ricky H. Jones[4], Curtis P. Langlotz[3], Andrew Y. Ng[1‡], Matthew P. Lungren[3‡]

1 Department of Computer Science, Stanford University, Stanford, California, United States of America, **2** Quantitative Sciences Unit, Department of Medicine, Stanford University, Stanford, California, United States of America, **3** Department of Radiology, Stanford University, Stanford, California, United States of America, **4** Department of Orthopedic Surgery, Stanford University, Stanford, California, United States of America

☯ These authors contributed equally to this work.
‡ These authors are joint senior authors on this work.
* nbien@stanford.edu

OPEN ACCESS

Citation: Bien N, Rajpurkar P, Ball RL, Irvin J, Park A, Jones E, et al. (2018) Deep-learning-assisted diagnosis for knee magnetic resonance imaging: Development and retrospective validation of MRNet. PLoS Med 15(11): e1002699. https://doi.org/10.1371/journal.pmed.1002699

Academic Editor: Suchi Saria, Johns Hopkins University, UNITED STATES

Received: June 2, 2018

Accepted: October 23, 2018

Published: November 27, 2018

Data Availability Statement: The data from Stanford University Medical Center used in this study are available at https://stanfordmlgroup.github.io/projects/MRNet to users who accept a Dataset Research Use Agreement. Code for replicating these findings is provided as Supporting Information (S1 Code and S2 Code).

Funding: The authors received no specific funding for this work.

Competing interests: I have read the journal's policy and the authors of this manuscript have the

Abstract

Background

Magnetic resonance imaging (MRI) of the knee is the preferred method for diagnosing knee injuries. However, interpretation of knee MRI is time-intensive and subject to diagnostic error and variability. An automated system for interpreting knee MRI could prioritize high-risk patients and assist clinicians in making diagnoses. Deep learning methods, in being able to automatically learn layers of features, are well suited for modeling the complex relationships between medical images and their interpretations. In this study we developed a deep learning model for detecting general abnormalities and specific diagnoses (anterior cruciate ligament [ACL] tears and meniscal tears) on knee MRI exams. We then measured the effect of providing the model's predictions to clinical experts during interpretation.

Methods and findings

Our dataset consisted of 1,370 knee MRI exams performed at Stanford University Medical Center between January 1, 2001, and December 31, 2012 (mean age 38.0 years; 569 [41.5%] female patients). The majority vote of 3 musculoskeletal radiologists established reference standard labels on an internal validation set of 120 exams. We developed MRNet, a convolutional neural network for classifying MRI series and combined predictions from 3 series per exam using logistic regression. In detecting abnormalities, ACL tears, and meniscal tears, this model achieved area under the receiver operating characteristic curve (AUC) values of 0.937 (95% CI 0.895, 0.980), 0.965 (95% CI 0.938, 0.993), and 0.847 (95% CI 0.780, 0.914), respectively, on the internal validation set. We also obtained a public dataset

Medical Image Analysis 35 (2017) 303–312

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

Large scale deep learning for computer aided detection of mammographic lesions

Thijs Kooi[a,*], Geert Litjens[a], Bram van Ginneken[a], Albert Gubern-Mérida[a], Clara I. Sánchez[a], Ritse Mann[a], Ard den Heeten[b], Nico Karssemeijer[a]

[a] Diagnostic Image Analysis Group, Department of Radiology, Radboud University Medical Center, Nijmegen, The Netherlands
[b] Department of Radiology, University Medical Centre Amsterdam, Amsterdam, The Netherlands

ARTICLE INFO

Article history:
Received 11 February 2016
Revised 12 July 2016
Accepted 20 July 2016
Available online 2 August 2016

Keywords:
Computer aided detection
Mammography
Deep learning
Machine learning
Breast cancer
Convolutional neural networks

ABSTRACT

Recent advances in machine learning yielded new techniques to train deep neural networks, which resulted in highly successful applications in many pattern recognition tasks such as object detection and speech recognition. In this paper we provide a head-to-head comparison between a state-of-the art in mammography CAD system, relying on a manually designed feature set and a Convolutional Neural Network (CNN), aiming for a system that can ultimately read mammograms independently. Both systems are trained on a large data set of around 45,000 images and results show the CNN outperforms the traditional CAD system at low sensitivity and performs comparable at high sensitivity. We subsequently investigate to what extent features such as location and patient information and commonly used manual features can still complement the network and see improvements at high specificity over the CNN especially with location and context features, which contain information not available to the CNN. Additionally, a reader study was performed, where the network was compared to certified screening radiologists on a patch level and we found no significant difference between the network and the readers.

1. Introduction

Nearly 40 million mammographic exams are performed in the US alone on a yearly basis, arising predominantly from screening programs implemented to detect breast cancer at an early stage, which has been shown to increase chances of survival (Tabar et al., 2003; Broeders et al., 2012). Similar programs have been implemented in many western countries. All this data has to be inspected for signs of cancer by one or more experienced readers which is a time consuming, costly and most importantly error prone endeavor. Striving for optimal health care, Computer Aided Detection and Diagnosis (CAD) (Giger et al., 2001; Doi, 2007; 2005; van Ginneken et al., 2011) systems are being developed and are currently widely employed as a second reader (Rao et al., 2010; Malich et al., 2006), with numbers from the US going up to 70% of all screening studies in hospital facilities and 85% in private institutions (Rao et al., 2010). Computers do not suffer from drops in concentration, are consistent when presented with the same input data and can potentially be trained with an incredible amount of training samples, vastly more than any radiologist will experience in his lifetime.

Until recently, the effectiveness of CAD systems and many other pattern recognition applications depended on meticulously hand-crafted features, topped off with a learning algorithm to map it to a decision variable. Radiologists are often consulted in the process of feature design and features such as the contrast of the lesion, spiculation patterns and the sharpness of the border are used, in the case of mammography. These feature transformations provide a platform to instill task-specific, a-priori knowledge, but cause a large bias towards how we humans think the task is performed. Since the inception of Artificial Intelligence (AI) as a scientific discipline, research has seen a shift from rule-based, problem specific solutions to increasingly generic, problem agnostic methods based on learning, of which *deep learning* (Bengio, 2009; Bengio et al., 2013; Schmidhuber, 2015; LeCun et al., 2015) is its most recent manifestation. Directly distilling information from training samples, rather than the domain expert, deep learning allows us to optimally exploit the ever increasing amounts of data and reduce human bias. For many pattern recognition tasks, this has proven to be successful to such an extent that systems are now reaching human or even superhuman performance (Cireşan et al., 2012; Mnih et al., 2015; He et al., 2015).

* Corresponding author.
E-mail address: thijs.kooi@radboudumc.nl, email@thijskooi.com (T. Kooi).

http://dx.doi.org/10.1016/j.media.2016.07.007
1361-8415/

ARTICLES
https://doi.org/10.1038/s41591-018-0107-6
nature medicine

Clinically applicable deep learning for diagnosis and referral in retinal disease

Jeffrey De Fauw[1], Joseph R. Ledsam[1], Bernardino Romera-Paredes[1], Stanislav Nikolov[1], Nenad Tomasev[1], Sam Blackwell[1], Harry Askham[1], Xavier Glorot[1], Brendan O'Donoghue[1], Daniel Visentin[1], George van den Driessche[1], Balaji Lakshminarayanan[1], Clemens Meyer[1], Faith Mackinder[1], Simon Bouton[1], Kareem Ayoub[1], Reena Chopra[2], Dominic King[1], Alan Karthikesalingam[1], Cían O. Hughes[1,3], Rosalind Raine[3], Julian Hughes[2], Dawn A. Sim[2], Catherine Egan[2], Adnan Tufail[2], Hugh Montgomery[3], Demis Hassabis[1], Geraint Rees[3], Trevor Back[1], Peng T. Khaw[2], Mustafa Suleyman[1], Julien Cornebise[1,3,4], Pearse A. Keane[2,4*] and Olaf Ronneberger[1,4*]

The volume and complexity of diagnostic imaging is increasing at a pace faster than the availability of human expertise to interpret it. Artificial intelligence has shown great promise in classifying two-dimensional photographs of some common diseases and typically relies on databases of millions of annotated images. Until now, the challenge of reaching the performance of expert clinicians in a real-world clinical pathway with three-dimensional diagnostic scans has remained unsolved. Here, we apply a novel deep learning architecture to a clinically heterogeneous set of three-dimensional optical coherence tomography scans from patients referred to a major eye hospital. We demonstrate performance in making a referral recommendation that reaches or exceeds that of experts on a range of sight-threatening retinal diseases after training on only 14,884 scans. Moreover, we demonstrate that the tissue segmentations produced by our architecture act as a device-independent representation; referral accuracy is maintained when using tissue segmentations from a different type of device. Our work removes previous barriers to wider clinical use without prohibitive training data requirements across multiple pathologies in a real-world setting.

Medical imaging is expanding globally at an unprecedented rate[1,2], leading to an ever-expanding quantity of data that requires human expertise and judgement to interpret and triage. In many clinical specialities there is a relative shortage of this expertise to provide timely diagnosis and referral. For example, in ophthalmology, the widespread availability of optical coherence tomography (OCT) has not been matched by the availability of expert humans to interpret scans and refer patients to the appropriate clinical care[3]. This problem is exacerbated by the marked increase in prevalence of sight-threatening diseases for which OCT is the gold standard of initial assessment[4–7].

Artificial intelligence (AI) provides a promising solution for such medical image interpretation and triage, but despite recent breakthrough studies in which expert-level performance on two-dimensional photographs in preclinical settings has been demonstrated[8,9], prospective clinical application of this technology remains stymied by three key challenges. First, AI (typically trained on hundreds of thousands of examples from one canonical dataset) must generalize to new populations and devices without a substantial loss of performance, and without prohibitive data requirements for retraining. Second, AI tools must be applicable to real-world scans, problems and pathways, and designed for clinical evaluation and deployment. Finally, AI tools must match or exceed the performance of human experts in such real-world situations. Recent work applying AI to OCT has shown promise in resolving some of these criteria in isolation, but has not yet shown clinical applicability by resolving all three.

Results

Clinical application and AI architecture. We developed our architecture in the challenging context of OCT imaging for ophthalmology. We tested this approach for patient triage in a typical ophthalmology clinical referral pathway, comprising more than 50 common diagnoses for which OCT provides the definitive imaging modality (Supplementary Table 1). OCT is a three-dimensional volumetric medical imaging technique analogous to three-dimensional ultrasonography but measuring the reflection of near-infrared light rather than sound waves at a resolution for living human tissue of ~5 μm[10]. OCT is now one of the most common imaging procedures with 5.35 million OCT scans performed in the US Medicare population in 2014 alone (see https://www.cms.gov/Research-Statistics-Data-and-Systems/Statistics-Trends-and-Reports/Medicare-Provider-Charge-Data/Physician-and-Other-Supplier.html). It has been widely adopted across the UK National Health Service (NHS) for comprehensive initial assessment and triage of patients requiring rapid non-elective assessment of acute and chronic sight loss. Rapid access 'virtual' OCT clinics have become the standard of care[11,12]. In such clinics, expert clinicians interpret the OCT and clinical history to diagnose and triage patients with

[1]DeepMind, London, UK. [2]NIHR Biomedical Research Centre at Moorfields Eye Hospital and UCL Institute of Ophthalmology, London, UK. [3]Present address: University College London, London, UK. [4]These authors contributed equally: Julien Cornebise, Pearse A. Keane, Olaf Ronneberger.
*e-mail: pearse.keane@moorfields.nhs.uk; olafr@deepmind.com

npj | Digital Medicine
www.nature.com/npjdigitalmed

ARTICLE OPEN

Advanced machine learning in action: identification of intracranial hemorrhage on computed tomography scans of the head with clinical workflow integration

Mohammad R. Arbabshirani[1], Brandon K. Fornwalt[1,2], Gino J. Mongelluzzo[1], Jonathan D. Suever[1,2], Brandon D. Geise[1], Aalpen A. Patel[1,2] and Gregory J. Moore[1]

Intracranial hemorrhage (ICH) requires prompt diagnosis to optimize patient outcomes. We hypothesized that machine learning algorithms could automatically analyze computed tomography (CT) of the head, prioritize radiology worklists and reduce time to diagnosis of ICH. 46,583 head CTs (~2 million images) acquired from 2007–2017 were collected from several facilities across Geisinger. A deep convolutional neural network was trained on 37,074 studies and subsequently evaluated on 9499 unseen studies. The predictive model was implemented prospectively for 3 months to re-prioritize "routine" head CT studies as "stat" on realtime radiology worklists if an ICH was detected. Time to diagnosis was compared between the re-prioritized "stat" and "routine" studies. A neuroradiologist blinded to the study reviewed false positive studies to determine whether the dictating radiologist overlooked ICH. The model achieved an area under the ROC curve of 0.846 (0.837–0.856). During implementation, 94 of 347 "routine" studies were re-prioritized to "stat", and 60/94 had ICH identified by the radiologist. Five new cases of ICH were identified, and median time to diagnosis was significantly reduced ($p < 0.0001$) from 512 to 19 min. In particular, one outpatient with vague symptoms on anticoagulation was found to have an ICH which was treated promptly with reversal of anticoagulation, resulting in a good clinical outcome. Of the 34 false positives, the blinded over-reader identified four probable ICH cases overlooked in original interpretation. In conclusion, an artificial intelligence algorithm can prioritize radiology worklists to reduce time to diagnosis of new outpatient ICH by 96% and may also identify subtle ICH overlooked by radiologists. This demonstrates the positive impact of advanced machine learning in radiology workflow optimization.

npj Digital Medicine (2018)1:9 ; doi:10.1038/s41746-017-0015-z

INTRODUCTION

Intracranial hemorrhage (ICH) is a critical condition accounting for about 2 million strokes worldwide.[1] Hemorrhages can occur both within the brain parenchyma (intra-axial) or within the cranial vault but external to the brain parenchyma (extra-axial). Both intra-axial and extra-axial hemorrhage have significant clinical burden. For example, intra-axial hemorrhage affects approximately 40,000 to 67,000 patients per year in the United States[2,3] with a 30-day mortality rate of 47 percent.[4] Moreover, 46% of survivors of subarachnoid hemorrhage (a type of extra-axial hemorrhage) endure permanent cognitive impairment.[5,6] Hospital admissions of ICH have dramatically increased in the past decade probably due to increased life expectancy and poor blood pressure control.[7,8] Importantly, early diagnosis of ICH is of critical clinical importance since nearly half of the resulting mortality occurs in the first 24 h,[9] and earlier treatment likely improves outcomes.[10] Computed tomography (CT) of the head is the most widely used tool for diagnosing acute ICH, and the timing of diagnosis, therefore, depends on how quickly a head CT is both completed and subsequently interpreted by a clinician.

The interpretation time of radiological studies is highly dependent on the priority assigned to the exam by the ordering physician (for example "stat" vs "routine") and by patient status (inpatient vs. outpatient). Stat studies are typically interpreted within an hour (at our institution) while routine outpatient studies can take much longer based on the available radiology workforce. Therefore, detection of ICH in routine studies (especially those imaged in an outpatient setting) may be significantly delayed. ICH does occur in the outpatient setting, albeit with a lesser frequency than the inpatient or emergency department setting. For example, elderly outpatients on anticoagulation therapy experience higher risk of ICH.[11] Importantly, initial symptoms may be vague, prompting a non-emergent, routine head CT.

Automatic triage of imaging studies using computer algorithms has the potential to detect ICH earlier, ultimately leading to improved clinical outcomes. Such a quality improvement tool could be used to automatically manage the priority for interpretation of imaging studies with presumed ICH and help optimize radiology workflow. Machine learning and computer vision are among a suite of techniques for teaching computers to learn and detect patterns.

In particular, deep learning (a class of machine learning algorithms suitable for training large multi-layer neural networks) has been leveraged for a variety of automated classification tasks

[1]Geisinger, Department of Radiology, 100 N. Academy Avenue, Danville, PA 17822-2007, USA and [2]Geisinger, Department of Imaging Science and Innovation, 100 N. Academy Avenue, Danville, PA 17822-4400, USA
Correspondence: Aalpen A. Patel (aapatel@geisinger.edu)
Mohammad R. Arbabshirani and Brandon K. Fornwalt contributed equally to this work.

Received: 18 September 2017 Revised: 7 December 2017 Accepted: 12 December 2017
Published online: 04 April 2018

Published in partnership with the Scripps Translational Science Institute
npj nature partner journals

nature machine intelligence
ARTICLES
https://doi.org/10.1038/s42256-019-0019-2

Deep-learning cardiac motion analysis for human survival prediction

Ghalib A. Bello[1,8], Timothy J. W. Dawes[1,2,8], Jinming Duan[1,3], Carlo Biffi[1,3], Antonio de Marvao[1], Luke S. G. E. Howard[4], J. Simon R. Gibbs[2,4], Martin R. Wilkins[5], Stuart A. Cook[1,2,6,7], Daniel Rueckert[3] and Declan P. O'Regan[1*]

Motion analysis is used in computer vision to understand the behaviour of moving objects in sequences of images. Optimizing the interpretation of dynamic biological systems requires accurate and precise motion tracking as well as efficient representations of high-dimensional motion trajectories so that these can be used for prediction tasks. Here we use image sequences of the heart, acquired using cardiac magnetic resonance imaging, to create time-resolved three-dimensional segmentations using a fully convolutional network trained on anatomical shape priors. This dense motion model formed the input to a supervised denoising autoencoder (4Dsurvival), which is a hybrid network consisting of an autoencoder that learns a task-specific latent code representation trained on observed outcome data, yielding a latent representation optimized for survival prediction. To handle right-censored survival outcomes, our network used a Cox partial likelihood loss function. In a study of 302 patients, the predictive accuracy (quantified by Harrell's *C*-index) was significantly higher ($P = 0.0012$) for our model $C = 0.75$ (95% CI: 0.70–0.79) than the human benchmark of $C = 0.59$ (95% CI: 0.53–0.65). This work demonstrates how a complex computer vision task using high-dimensional medical image data can efficiently predict human survival.

Techniques for vision-based motion analysis aim to understand the behaviour of moving objects in image sequences[1]. In this domain, deep-learning architectures have achieved a wide range of competencies for object tracking, action recognition and semantic segmentation[2]. Making predictions about future events from the current state of a moving three-dimensional (3D) scene depends on learning correspondences between patterns of motion and subsequent outcomes. Such relationships are important in biological systems that exhibit complex spatio-temporal behaviour in response to stimuli or as a consequence of disease processes. Here we use recent advances in machine learning for visual processing tasks to develop a generalizable approach for modelling time-to-event outcomes from time-resolved 3D sensory input. We tested this on the challenging task of predicting survival due to heart disease through analysis of cardiac imaging.

The motion dynamics of the beating heart are a complex rhythmic pattern of nonlinear trajectories regulated by molecular, electrical and biophysical processes[3]. Heart failure is a disturbance of this coordinated activity characterized by adaptations in cardiac geometry and motion that lead to impaired organ perfusion[4]. For this prediction task, we studied patients diagnosed with pulmonary hypertension, characterized by right ventricular (RV) dysfunction, as this is a disease with high mortality where the choice of treatment depends on individual risk stratification[5]. Our input data were derived from cardiac magnetic resonance (CMR), which acquires imaging of the heart in any anatomical plane for dynamic assessment of function. While explicit measurements of performance obtained from myocardial motion tracking detect early contractile dysfunction and act as discriminators of different pathologies[6,7], we hypothesized that learned features of complex 3D cardiac motion would provide enhanced prognostic accuracy.

A major challenge for medical image analysis has been to automatically derive quantitative and clinically relevant information in patients with disease phenotypes. Our method employs a fully convolutional network (FCN) to learn a cardiac segmentation task from manually labelled priors. The outputs are smooth 3D renderings of frame-wise cardiac motion that are used as input data to a supervised denoising autoencoder (DAE) prediction network that we refer to as 4Dsurvival. The aim is to learn latent representations robust to noise and salient for survival prediction. We then compared our model to a benchmark of conventional human-derived volumetric indices and clinical risk factors in survival prediction.

Results

Baseline characteristics. Data from all 302 patients with incident pulmonary hypertension were included for analysis. Objective diagnosis was made according to haemodynamic and imaging criteria[5]. Patients were investigated between 2004 and 2017, and were followed-up until 27 November 2017 (median 371 days). All-cause mortality was 28% (85 of 302). Table 1 summarizes characteristics of the study sample at the date of diagnosis. No subjects' data were excluded.

Magnetic resonance image processing. Automatic segmentation of the ventricles from gated CMR images was performed for each slice position at each of 20 temporal phases, producing a total of 69,820 label maps for the cohort (Fig. 1a). Image registration was used to track the motion of corresponding anatomic points. Data for each subject were aligned, producing a dense model of cardiac

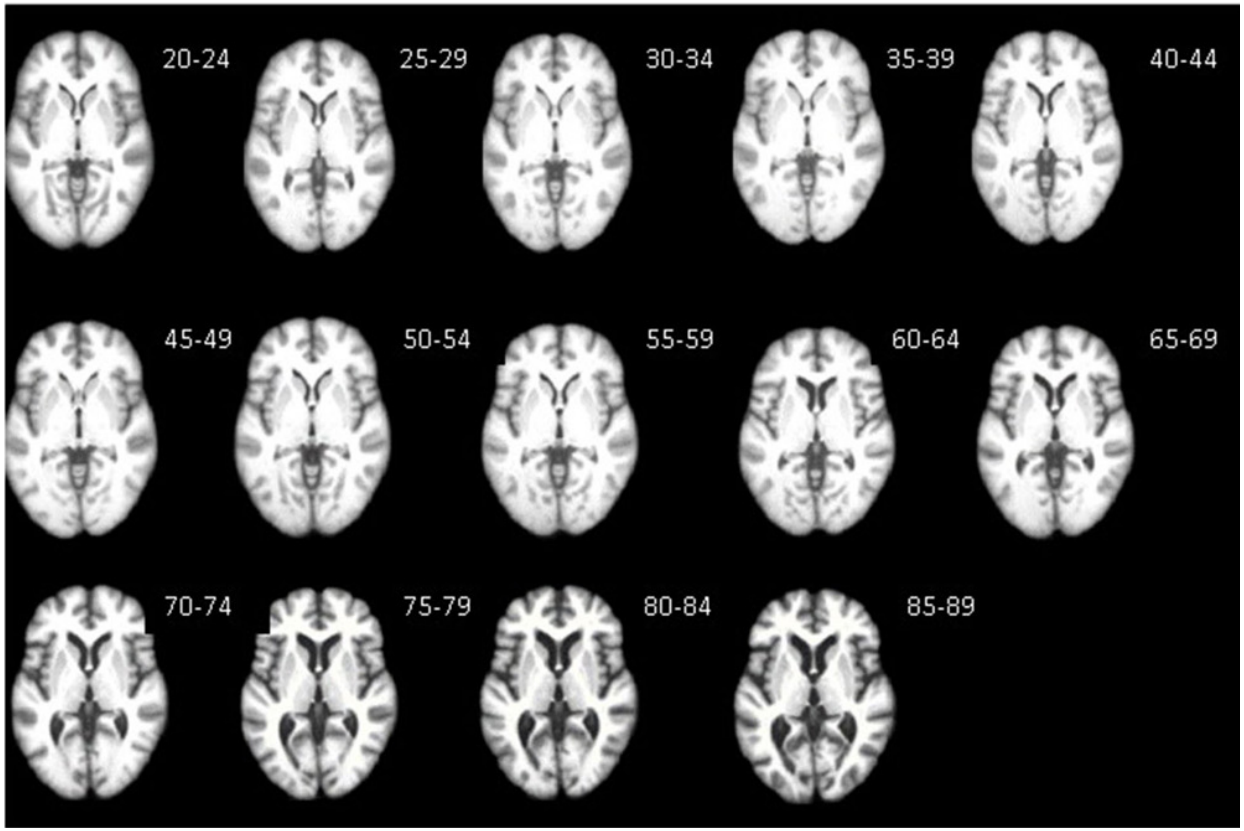
[1]MRC London Institute of Medical Sciences, Imperial College London, London, UK. [2]National Heart and Lung Institute, Imperial College London, London, UK. [3]Department of Computing, Imperial College London, London, UK. [4]Imperial College Healthcare NHS Trust, London, UK. [5]Division of Experimental Medicine, Department of Medicine, Imperial College London, London, UK. [6]National Heart Centre Singapore, Singapore, Singapore. [7]Duke-NUS Graduate Medical School, Singapore, Singapore. [8]These authors contributed equally: Ghalib A. Bello, Timothy J. W. Dawes. *e-mail: declan.oregan@imperial.ac.uk

Imperial College London

Case study: Brain Age Estimation from MR Images

Healthy Aging

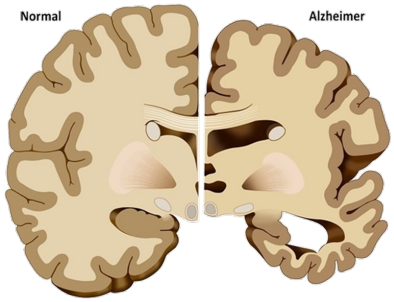
- Healthy aging results in the changing of brain volumes following a specific pattern



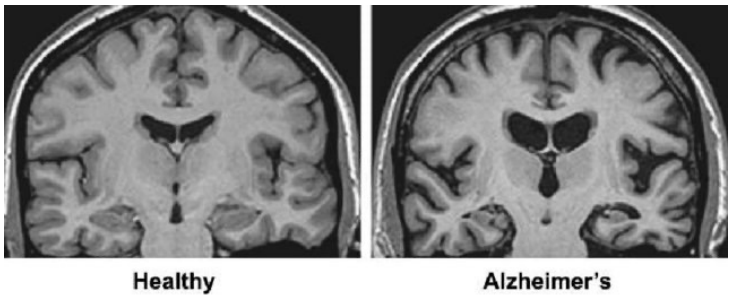
Brain aging as a potential biomarker

- Alzheimer's Disease:
  - connections between neurons stop working

  → atrophy → loss of brain volume

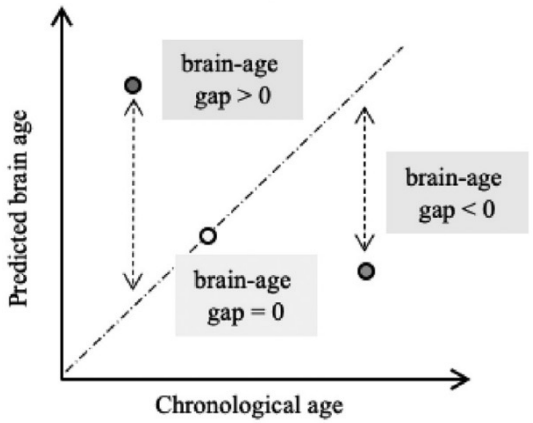


©ilusmedicall / Shutterstock.com



Juan Manuel Fernández Montenegro, Alzheimer's Disease Diagnosis Based on Cognitive Methods in Virtual Environments and Emotions Analysis, 2018

- Potential Biomarker: brain-age gap
  - Comparison of the predicted (biological) & the real (chronological) age of the subject

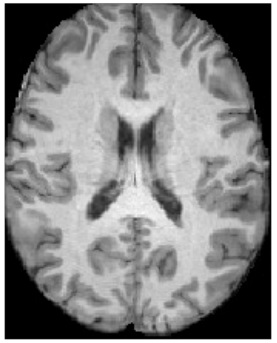


Baecker et al., Machine learning for brain age prediction: Introduction to methods and clinical applications, 2021

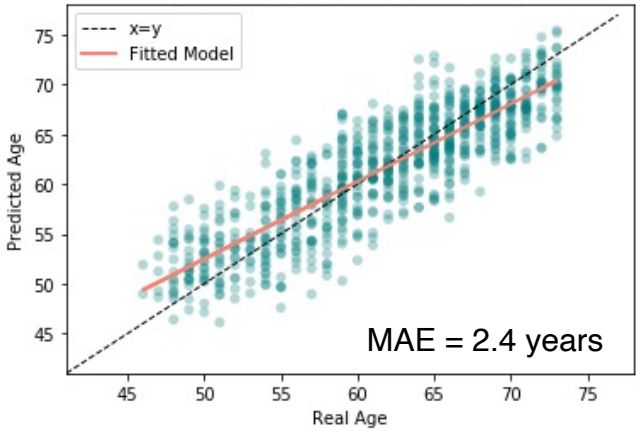
Machine Learning for Brain Age Regression

- MR Images are mainly used for age prediction (usually whole brain 2D or 3D images)
- Deep Learning models proved to be successful on the task
- Models trained on a dataset of MR images of healthy brains, estimating the expected chronological age of the subject
  - During training -> minimize error between chronological and real age (usually Mean Absolute Error – MAE)
  - During test time -> an estimated brain age larger than the subject's chronological age indicates accelerated aging, thus pointing to a possible AD patient
- State-of-the-art MAE as low as 2.2 years

Input: 3D structural MR Images
(here one slice is shown)



Model: 3D ResNet18- Uncorrected for Bias[a]



Bias Correction

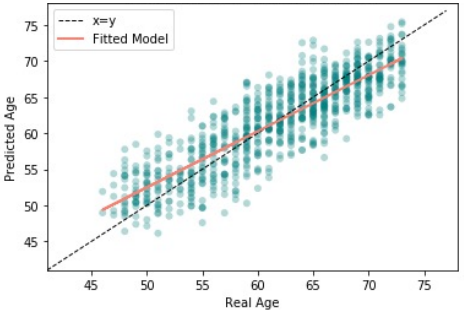
- There is a common bias when predicting age, possibly because of regression dilution
- Overestimation in younger subjects and underestimation in older subjects
- Recently, chronological age has been used as a covariate for predicting a bias corrected age

$$\Delta = \alpha * Y_{chronological} + \beta \quad (1)$$

- $Y_{chronological}$: real age of the subject.
- Δ: brain age delta function.
- The α and β represent the slope and intercept respectively and are then used for the estimation of the corrected predicted age from:

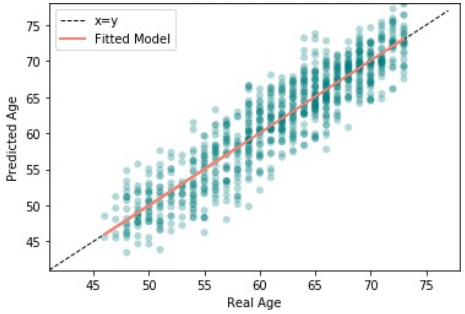
$$Y_{corrected} = Y_{predicted} - (\alpha * Y_{chronological} + \beta)$$

Before Bias Correction



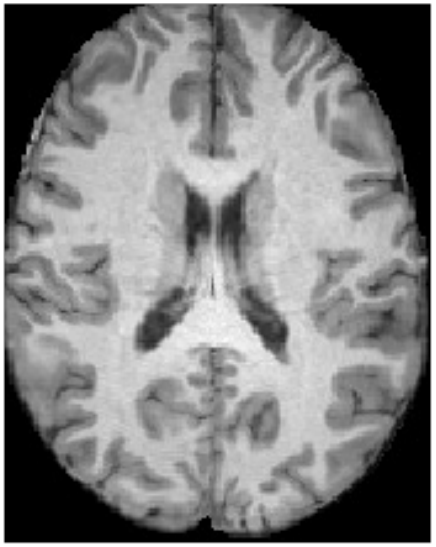
MAE = 2.64 years
R^2 score = 0.77

After Bias Correction

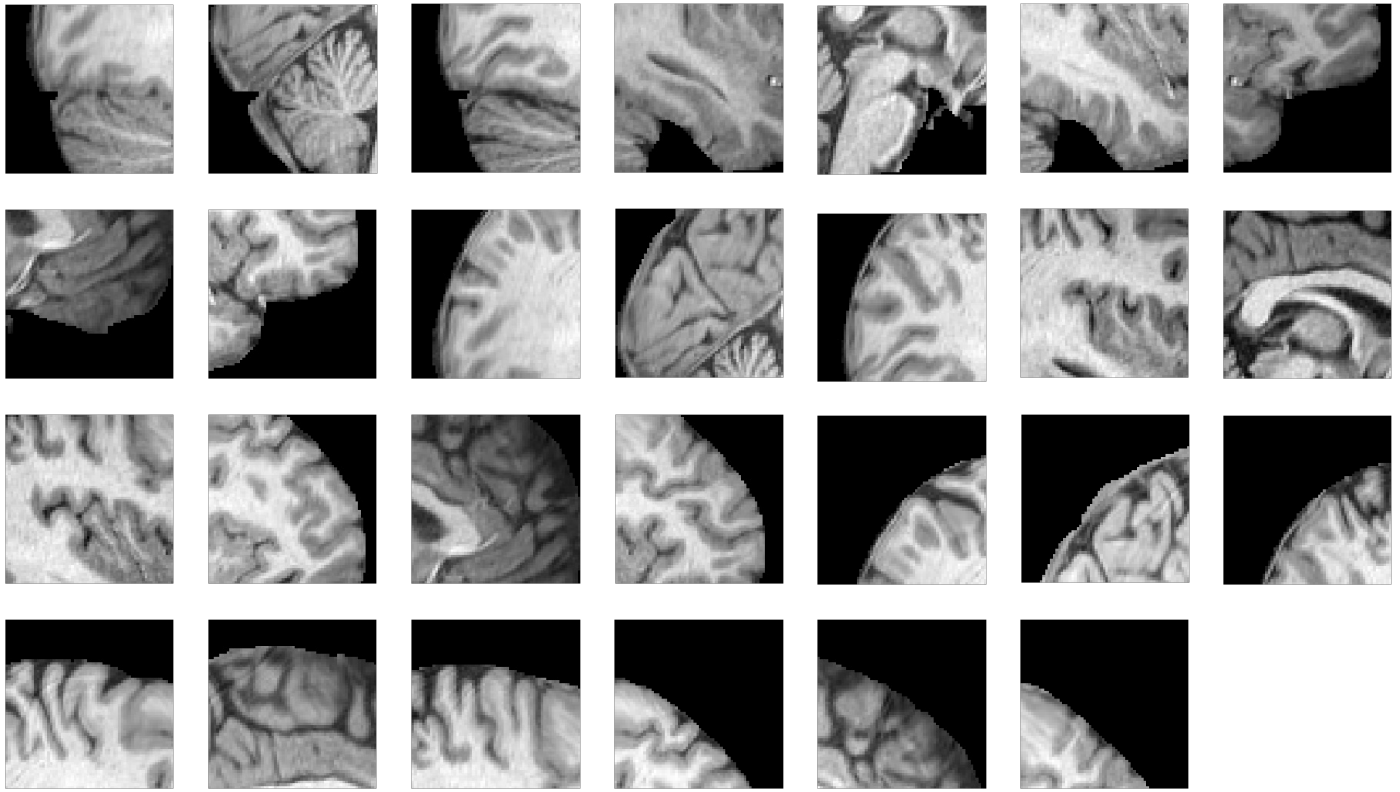


MAE = 2.43 years
R^2 score = 0.80

Patch Extraction

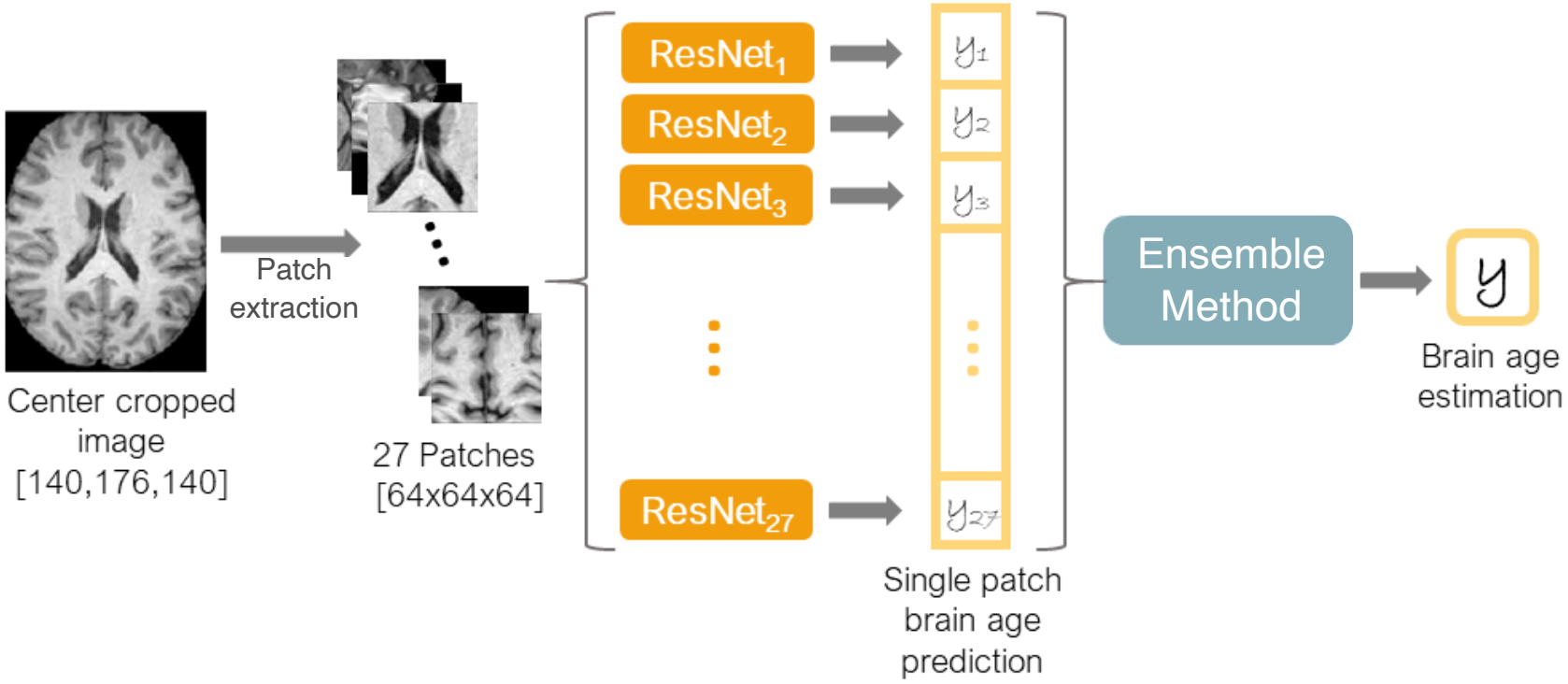


Patch Extraction



Bintsi, et al. "Patch-Based Brain Age Estimation from MR Images.", MLCN MICCAI 2020.

Patch-based Approach



Bintsi, et al. "Patch-Based Brain Age Estimation from MR Images.", MLCN MICCAI 2020.

Patch-based Predictions

- Mean Absolute Error (MAE) from around 2.45 to 4.2 years
- Colors indicate the performance of this single-patch model in terms of MAE

MAE of less than 3 years

MAE between 3 and 3.5 years

MAE between 3.5 and 4 years

MAE greater than 4 years

Bintsi, et al. "Patch-Based Brain Age Estimation from MR Images.", MLCN MICCAI 2020.

Ensemble Method

Averaging

$$y = \frac{1}{P}\sum_{i=1}^{P} y_i$$

P : subset of models
y_i : predictions

Results:

a. All patches: MAE = 2.66 yrs
   R^2 score = 0.78
b. Selected patches: MAE = 2.28 yrs
   R^2 score = 0.83

Linear Regression

$$y = w_0 + \sum_{i=1}^{P} w_i y_i$$

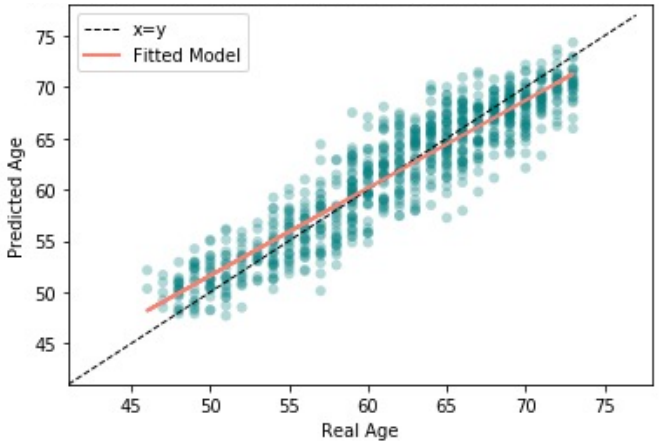
Results:

a. All patches: MAE = 2.13 yrs
   R^2 score = 0.85
a. Selected patches: MAE = 2. 16 yrs
   R^2 score = 0.85

Bintsi, et al. "Patch-Based Brain Age Estimation from MR Images.", MLCN MICCAI 2020.

Ensemble Method – Bias Correction

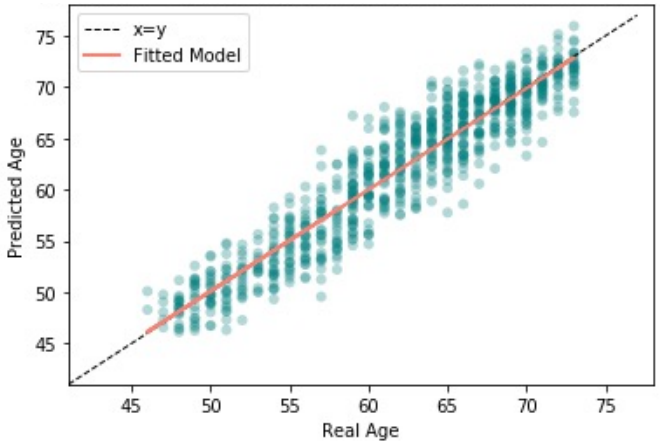
Ensemble = Linear Regression (all patches)
Uncorrected for Bias



MAE = 2.13 years

R^2 score = 0.85

Ensemble = Linear Regression (all patches)
Corrected for Bias



MAE = 1.96 years

R^2 score = 0.87

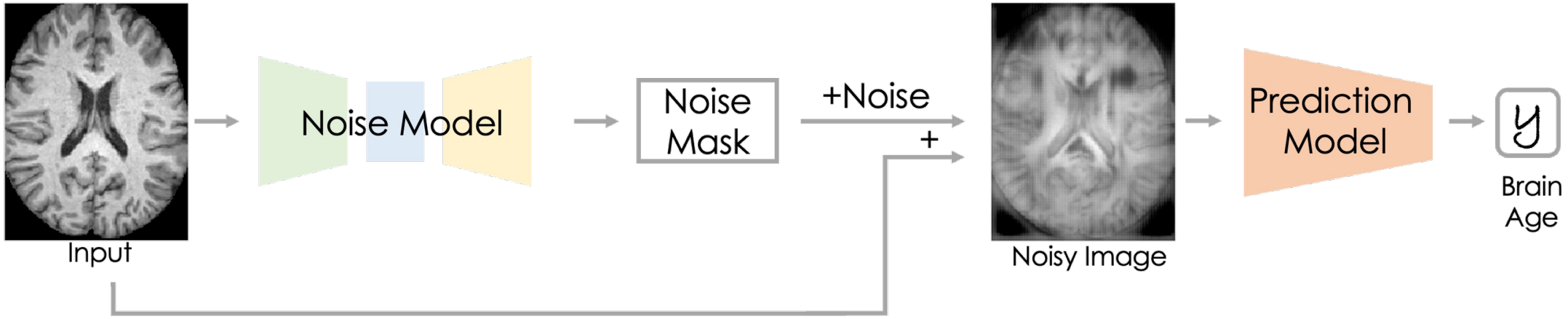
Bintsi, et al. "Patch-Based Brain Age Estimation from MR Images.", MLCN MICCAI 2020.

Conclusions

- We explore how patches of the brain perform the brain age estimation task.
- Patch-based Predictions: dependent on the region.
- Best performance: patches that include the ventricles and the hippocampus (MAE around 2.5 years).
- Ensemble method:
  - Averaging: major boost in the regression performance when selected patches are used (MAE = 2.26 years).
  - Linear Regression: provides state-of-the-art results (MAE = 2.13 years).

Bintsi, et al. "Patch-Based Brain Age Estimation from MR Images.", MLCN MICCAI 2020.

Importance Map Extraction

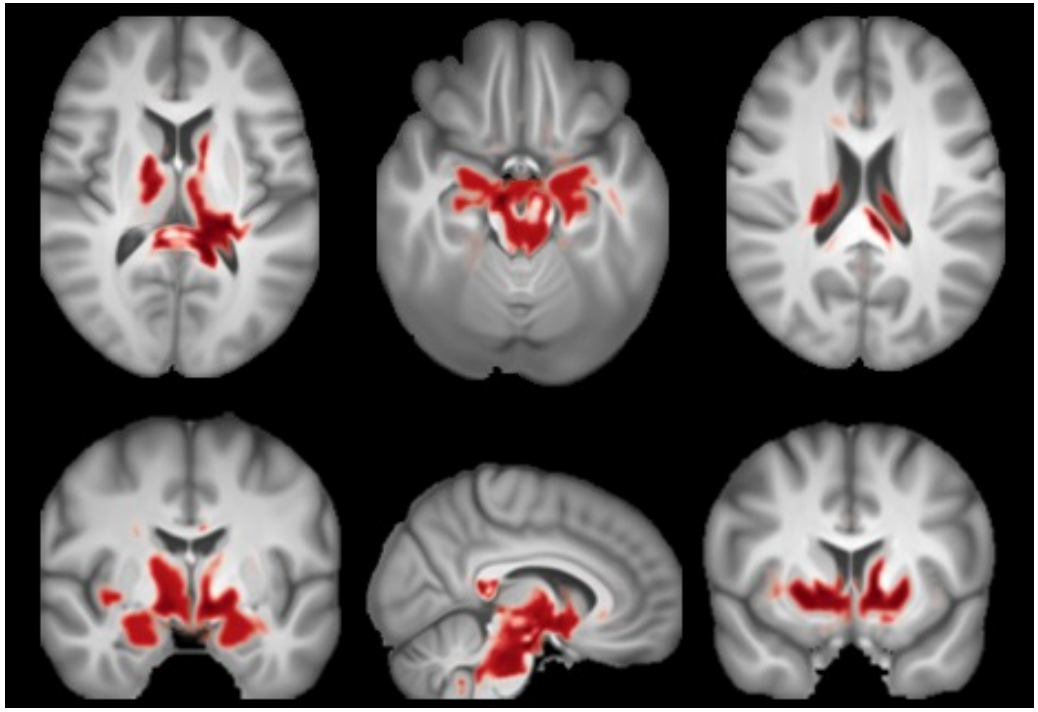


$$\mathcal{L} = \underbrace{(f_\theta(X) - y)^2}_{\text{prediction term}} - r \underbrace{log(f_\psi(X))}_{\text{noise term}}$$

Bintsi, et al. " Voxel-level Importance Maps for Interpretable Brain Age Estimation.", iMIMIC MICCAI 2021.

Population-Based Importance Maps

- Population-based importance maps:
  - Averaging of the importance maps for all subjects
  - Thresholding
- Relevant areas:
  - mesial temporal structures including hippocampus
  - parts of the ventricles
- Differences in cerebral cortex not captured. Why?
  - Age range not large enough
  - Images non-linearly registered



Bintsi, et al. " Voxel-level Importance Maps for Interpretable Brain Age Estimation.", iMIMIC MICCAI 2021.

Imperial College London

That’s all for now