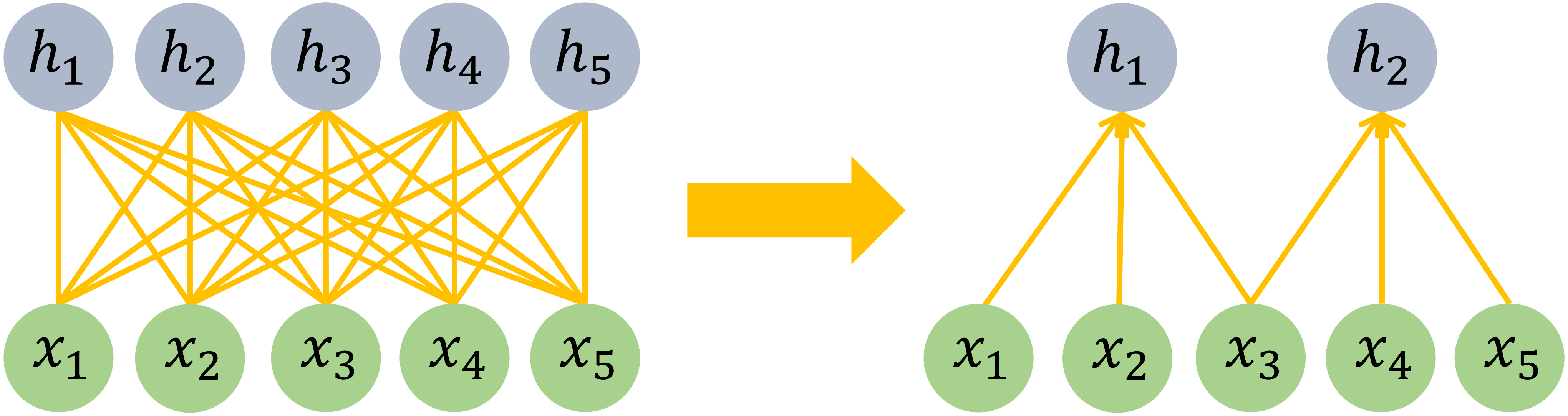
Deep Learning for Healthcare

Convolutional neural networks (CNN)

Jimeng Sun

ILLINOIS

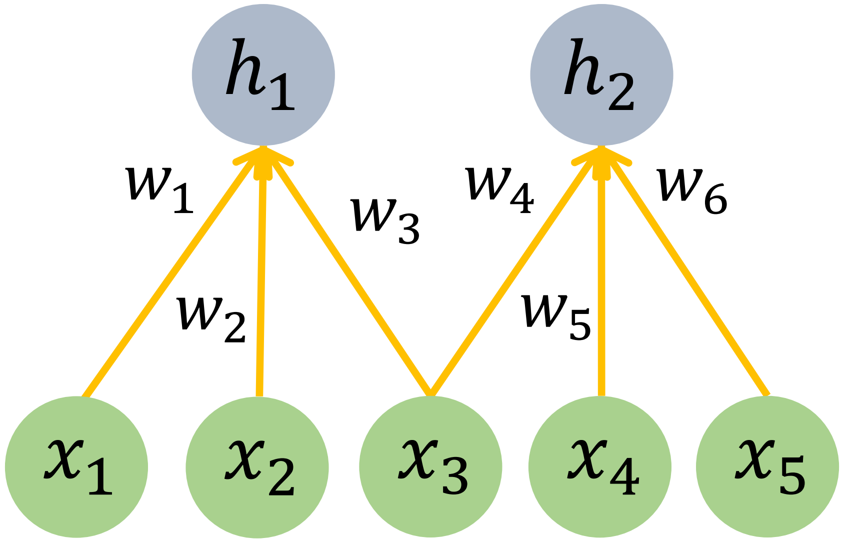
CONVOLUTION – LOCAL NETWORKS



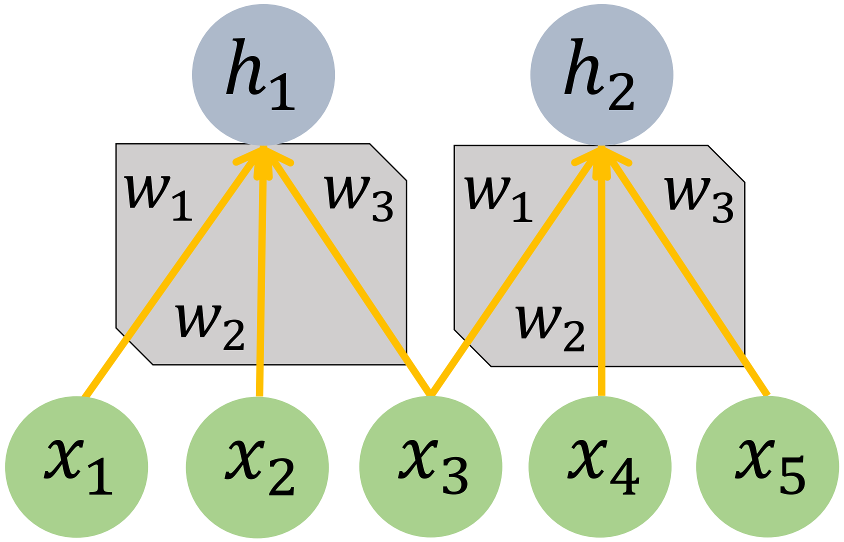
Fully Connected

Locally Connected

CONVOLUTION – WEIGHT SHARING

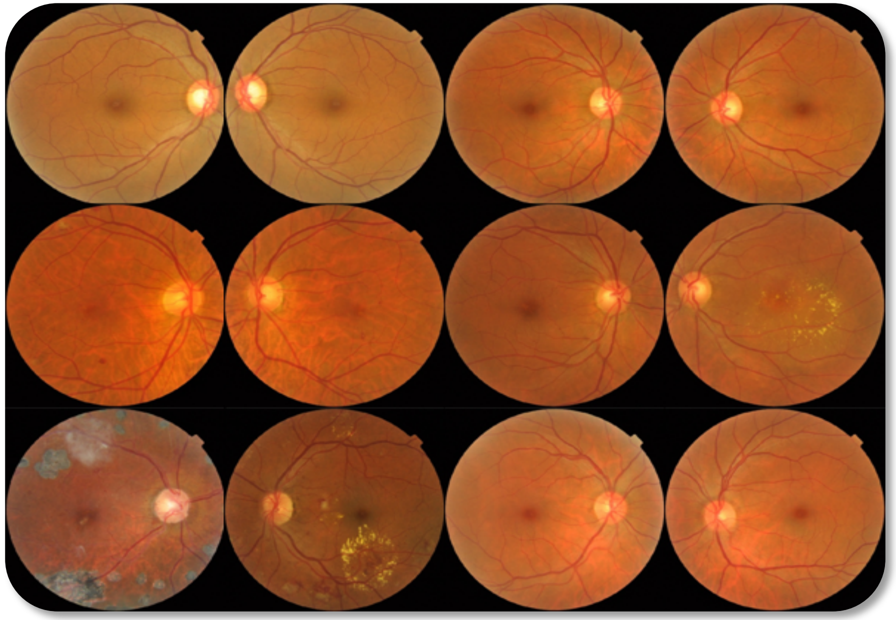


Locally Connected



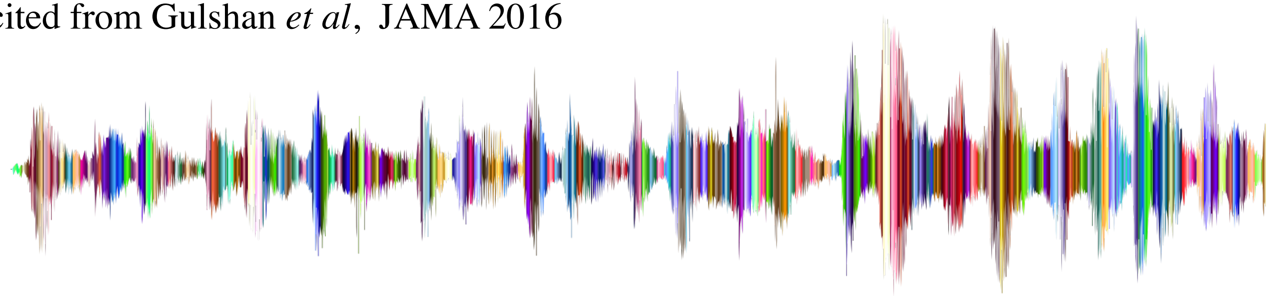
Locally Connected and Weight Sharing (Convolution)

POOLING – HANDLING DISTORTION

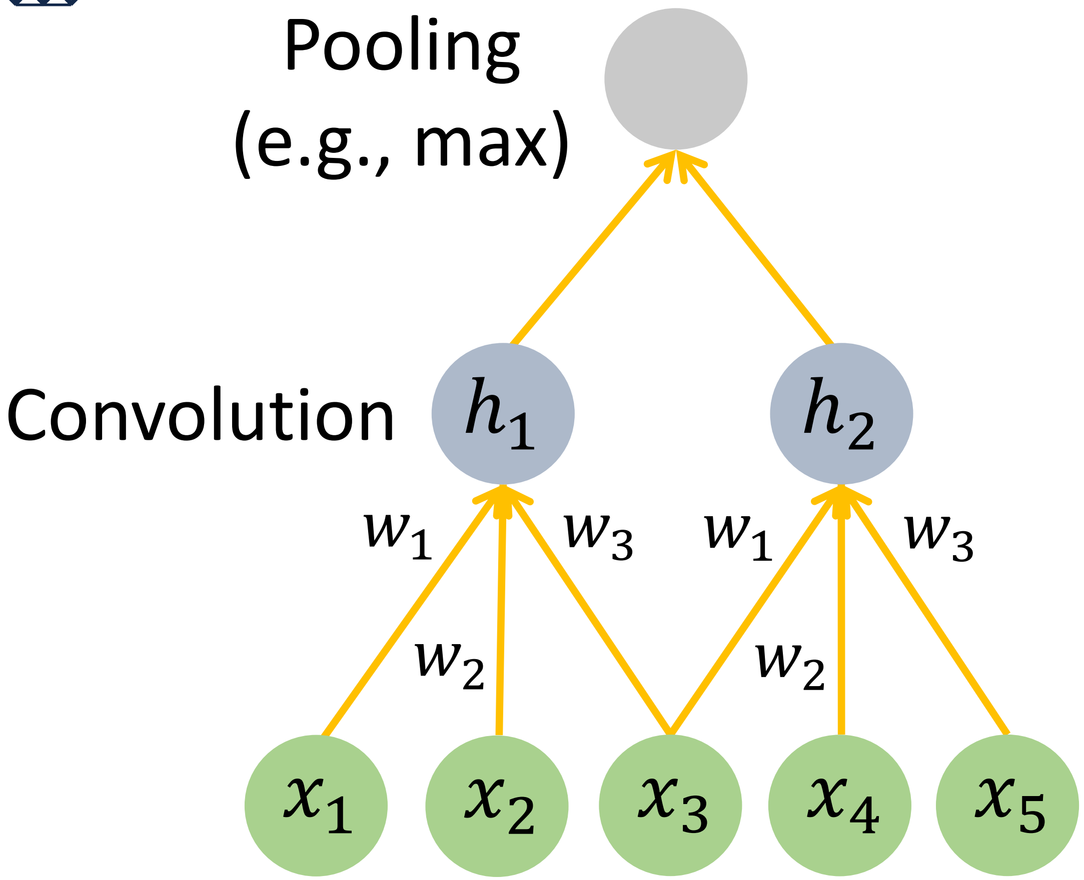


The medical image is cited from Gulshan *et al*, JAMA 2016

For grid-like data: main source of distortion is translation.



POOLING – HANDLING DISTORTION

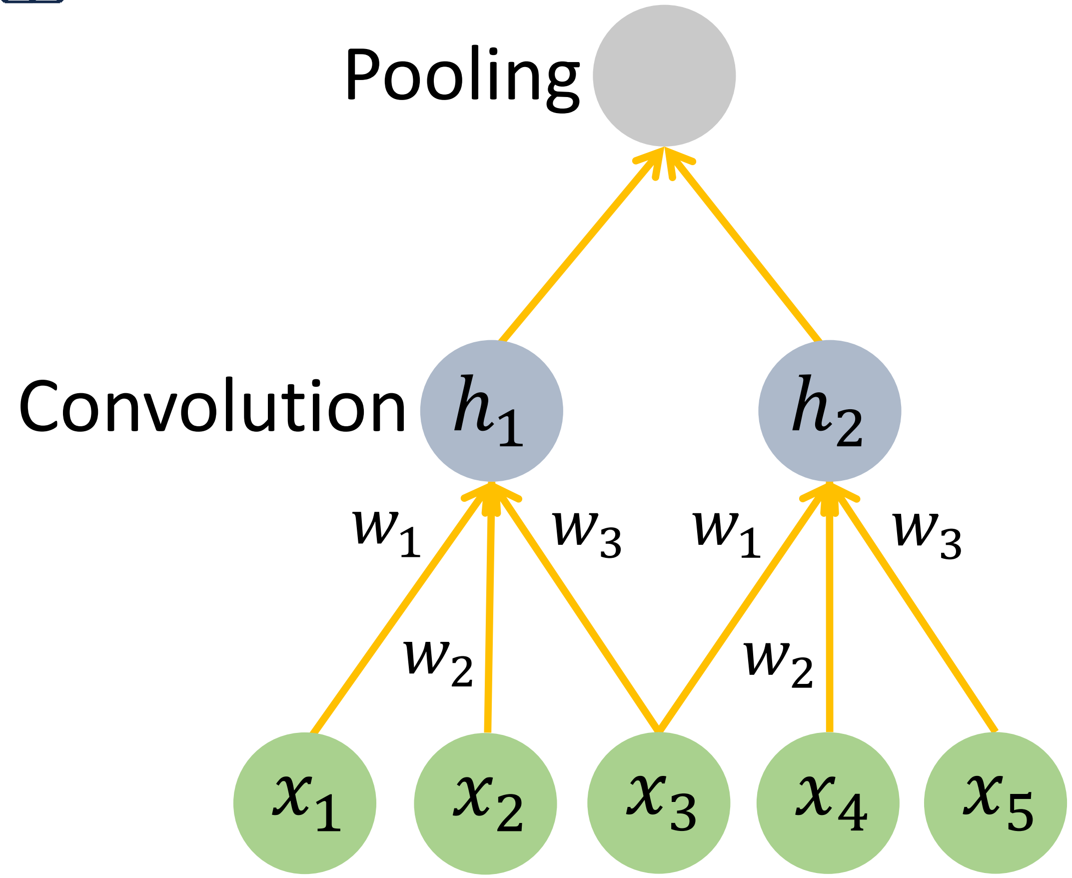


Translated by 2 positions

$$\boldsymbol{x} = [0,1,0,0,0] \rightarrow \max(\overset{h_1}{\mathrm{w}_2}, \overset{h_2}{0})$$

$$\boldsymbol{x} = [0,0,0,1,0] \rightarrow \max(0, \mathrm{w}_2)$$

CONVOLUTIONAL NEURAL NETWORKS (CNN)



Process data that has a grid-like structure (images, time series)

Utilize a specialized operation (convolution, pooling)

ADVANTAGES:
sparse interactions, parameter sharing, and translational invariance

ILLINOIS

BASIC CNN STRUCTURE

Fully connected (1000 output dimensions)

Fully connected (4096 dimensions)

Fully connected (4096 dimensions)

Max Pooling 3x3 filter, stride 2

CONV 256 3x3 filters, stride 1, padding 1

CONV 384 3x3 filters, stride 1, padding 1

CONV 384 3x3 filters, stride 1, padding 1

Max Pooling 3x3 filter, stride 2

CONV 256 5x5 filters

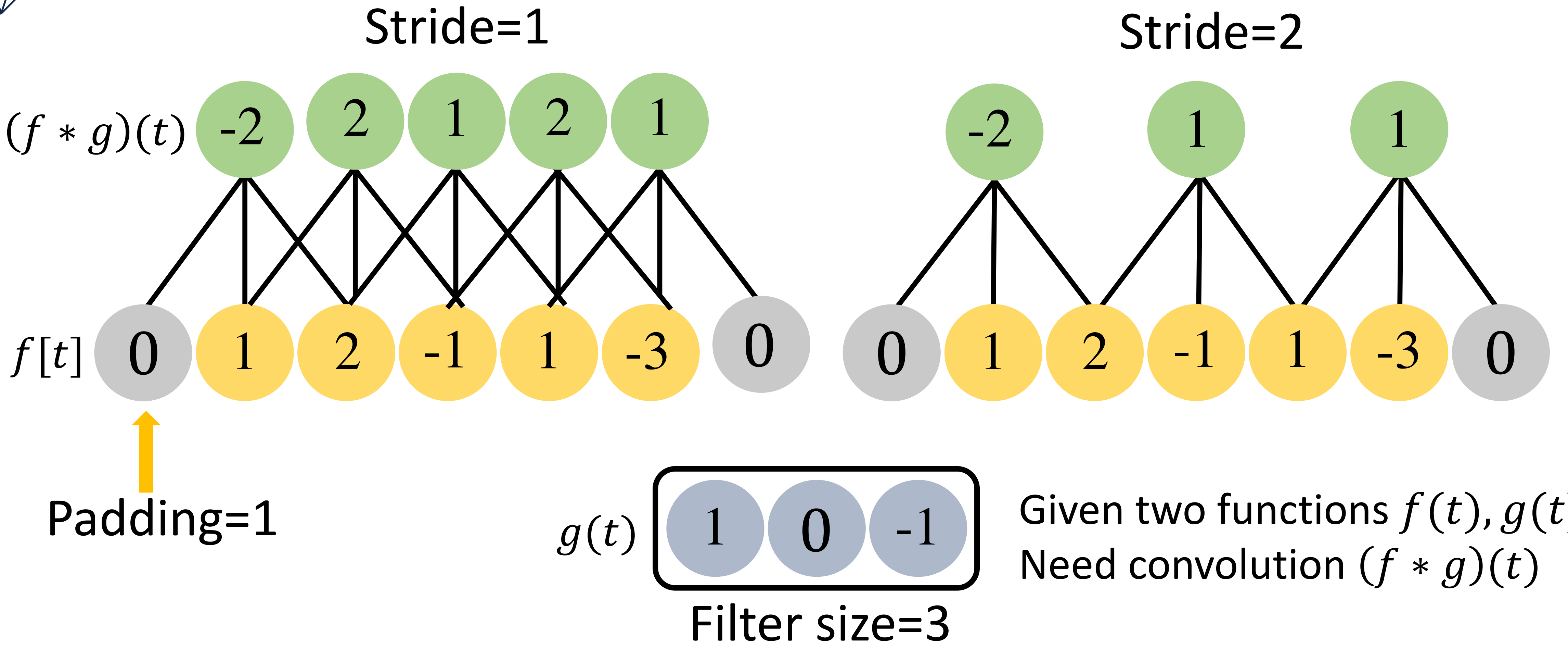
Max Pooling 3x3 filter, stride 2

CONV 96 11x11 filters , stride 4

227x227x3
input images

CONVOLUTION AND POOLING

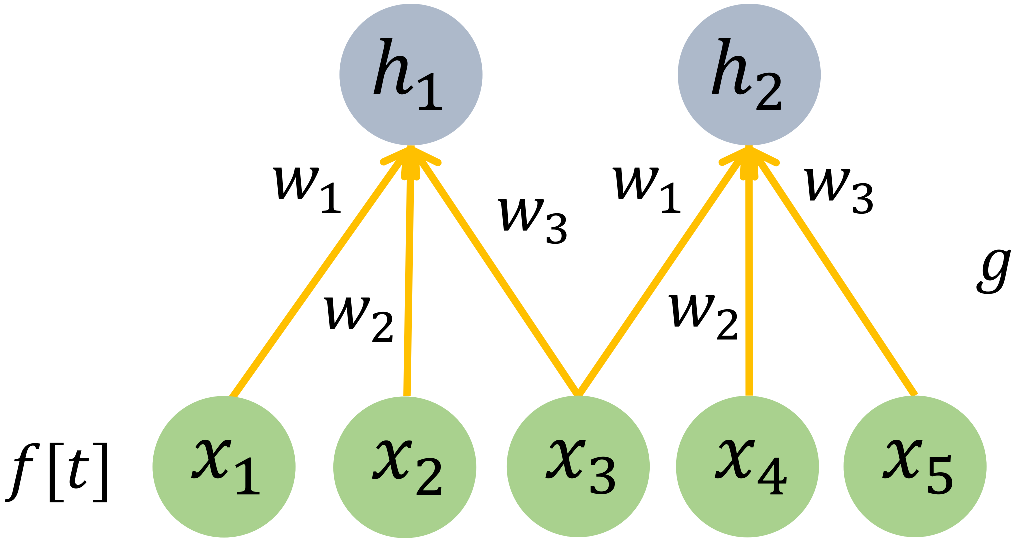
1-D CONVOLUTVION



Given two functions $f(t), g(t)$

Need convolution $(f * g)(t)$

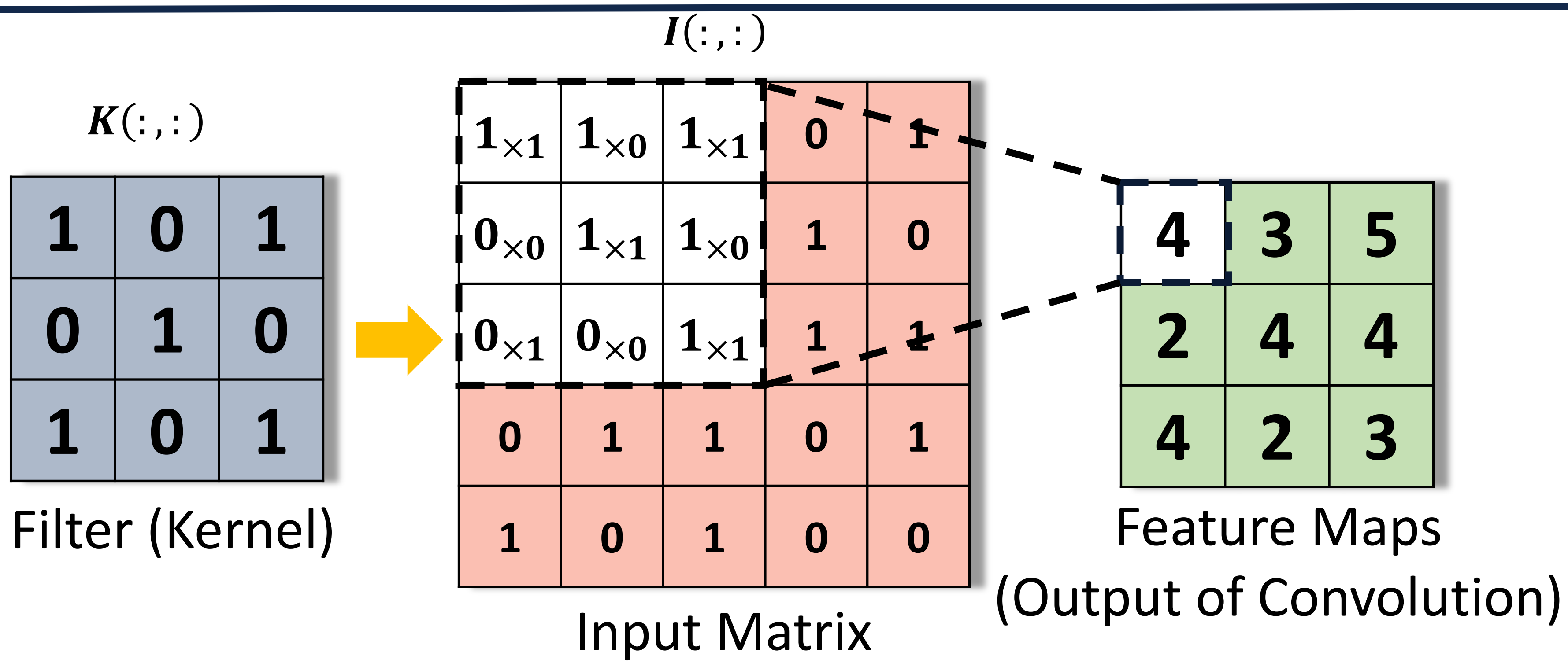
NEURAL NETWORK for 1-D CONVOLUTION



$$g[t] = [w_1, w_2, w_3]$$

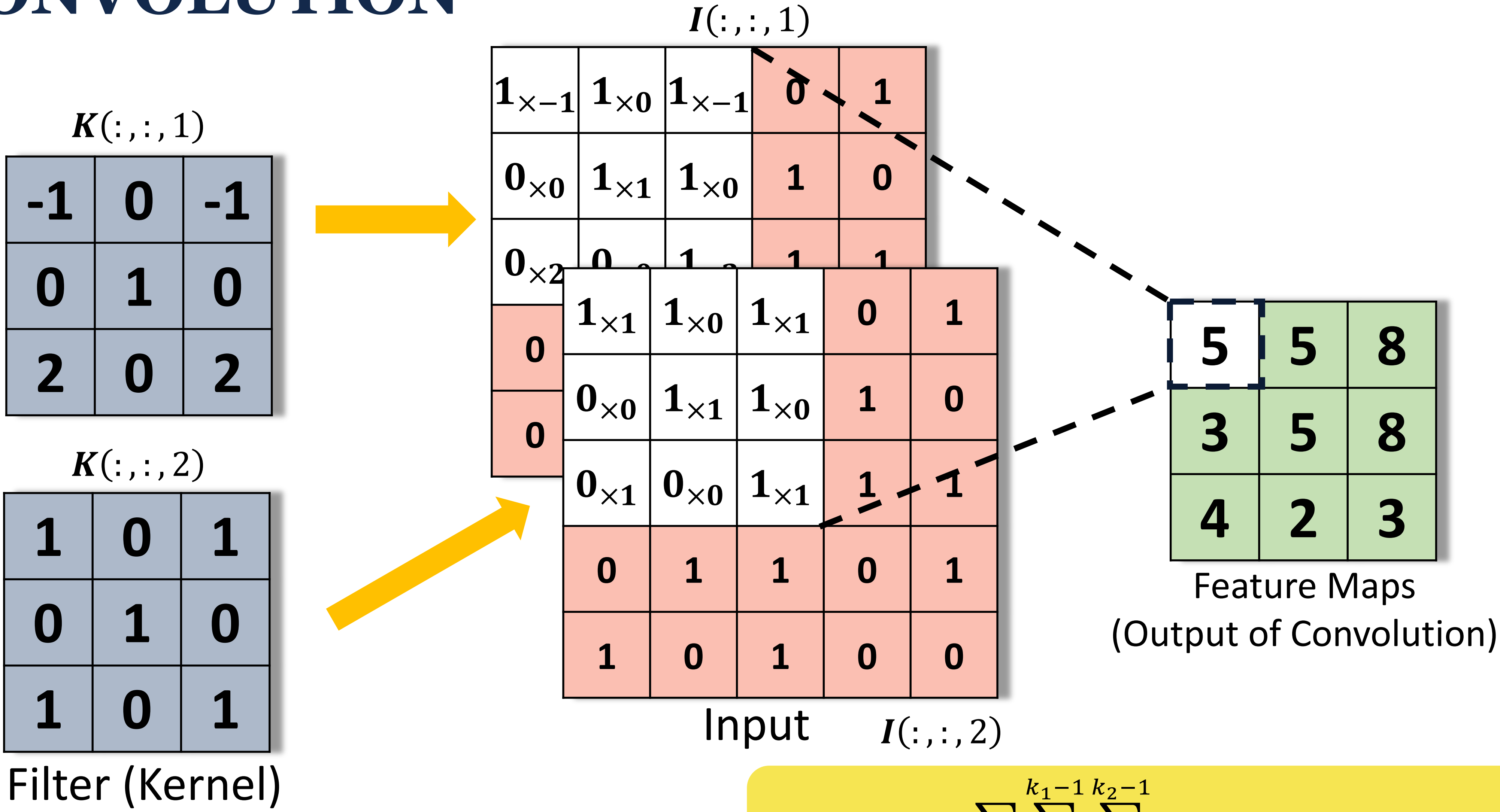
$$h_1 = (f * g)(t)_1 = w_1 x_1 + w_2 x_2 + w_3 x_3$$
$$h_2 = (f * g)(t)_2 = w_1 x_3 + w_2 x_4 + w_3 x_5$$

2-D CONVOLUTION OPERATION



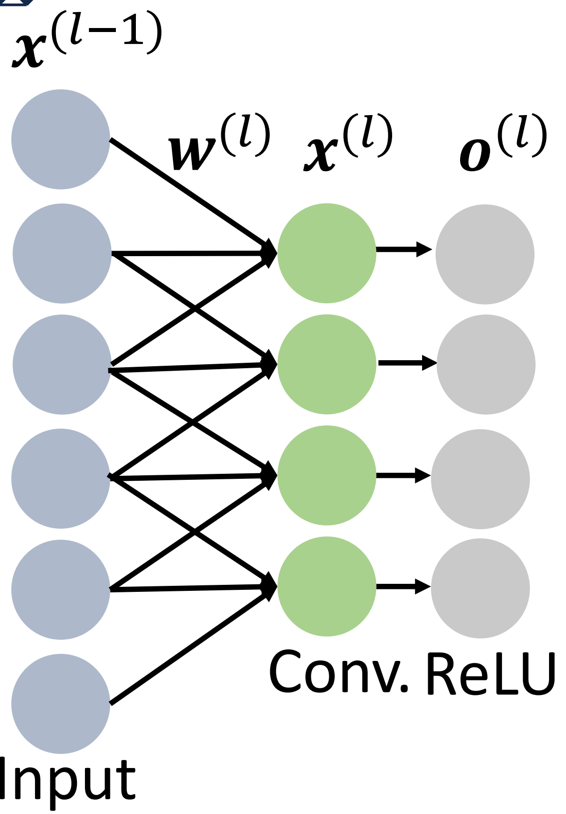
$$(\boldsymbol{I} * \boldsymbol{K})(i,j) = \sum_{m=0}^{k_1-1}\sum_{n=0}^{k_2-1} \boldsymbol{I}(i+m,j+n)\boldsymbol{K}(m,n)$$

3D CONVOLUTION

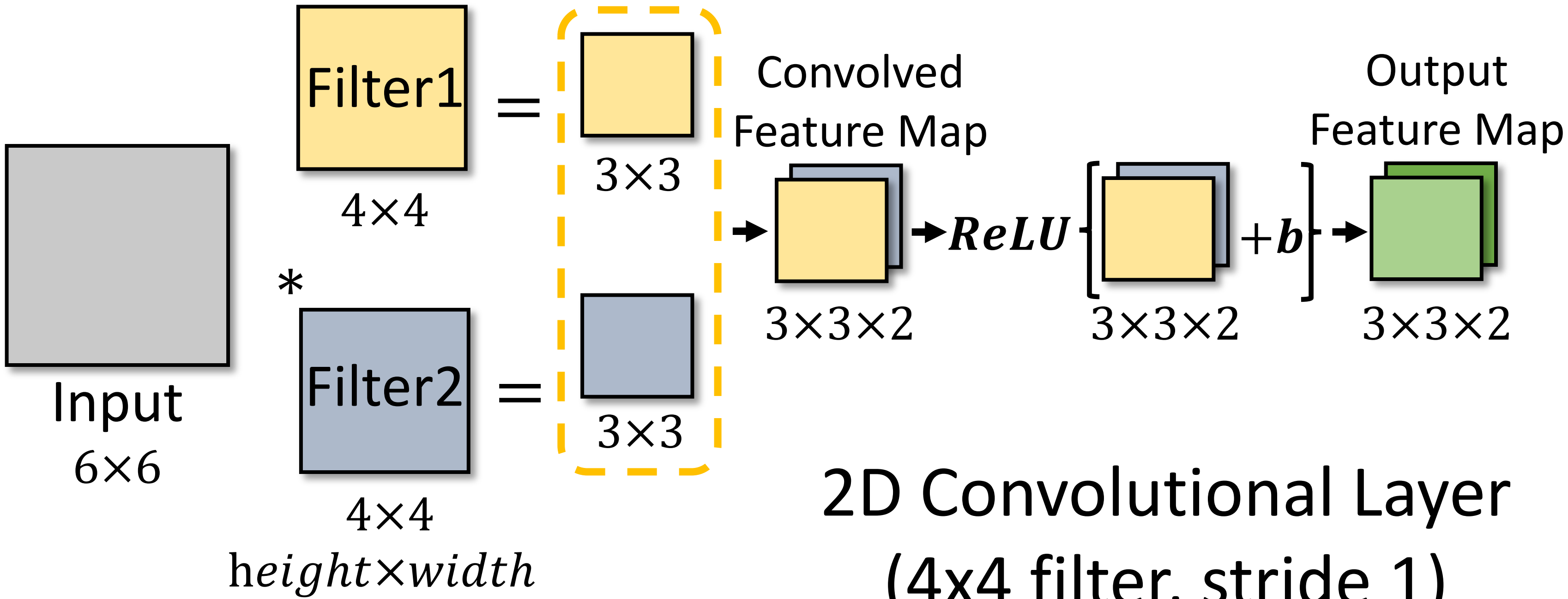


$$(\boldsymbol{I} * \boldsymbol{K})(i,j) = \sum_{d} \sum_{m=0}^{k_1-1} \sum_{n=0}^{k_2-1} \boldsymbol{I}(i+m, j+n, d)\boldsymbol{K}(m,n,d)$$

COMPLETE CONVOLUTION LAYER

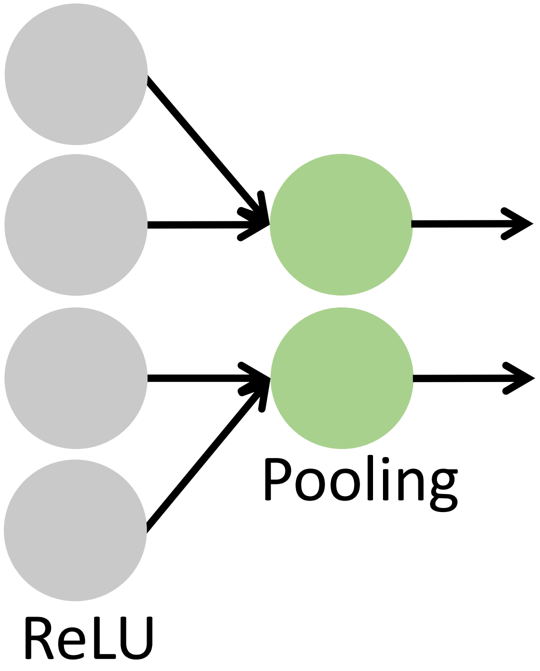


1D Convolutional Layer

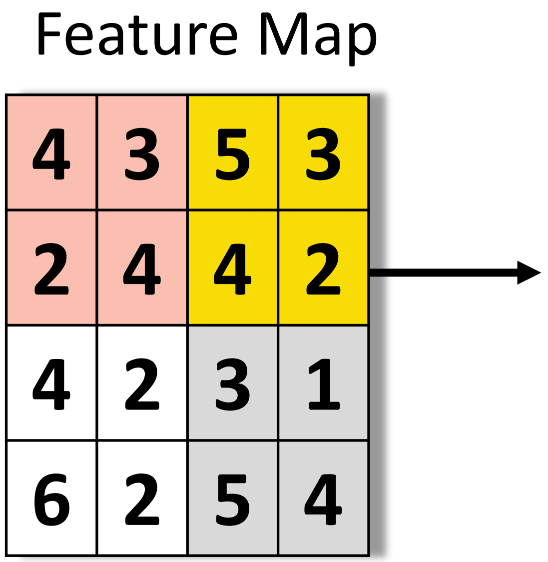


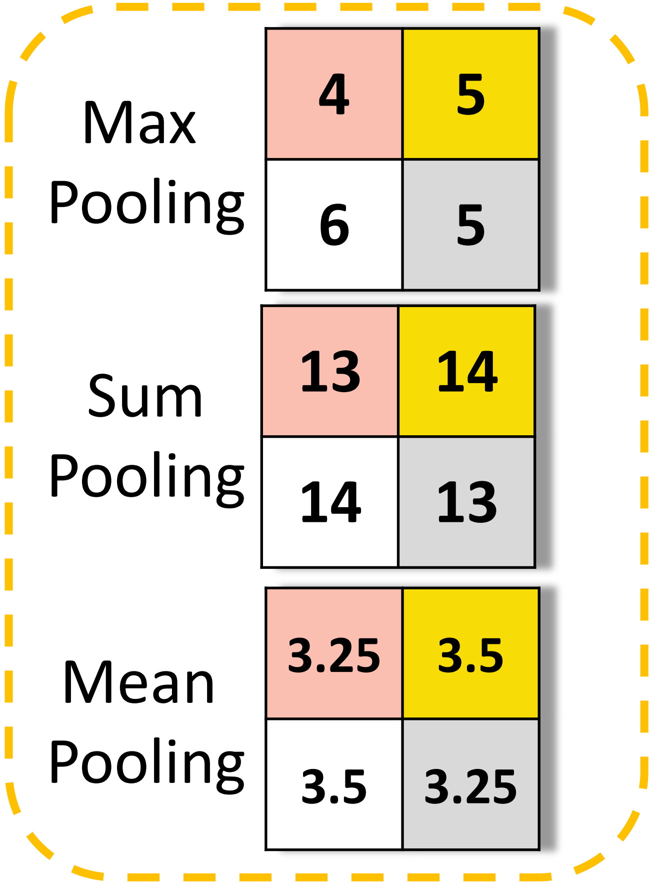
2D Convolutional Layer
(4x4 filter, stride 1)

POOLING LAYERS



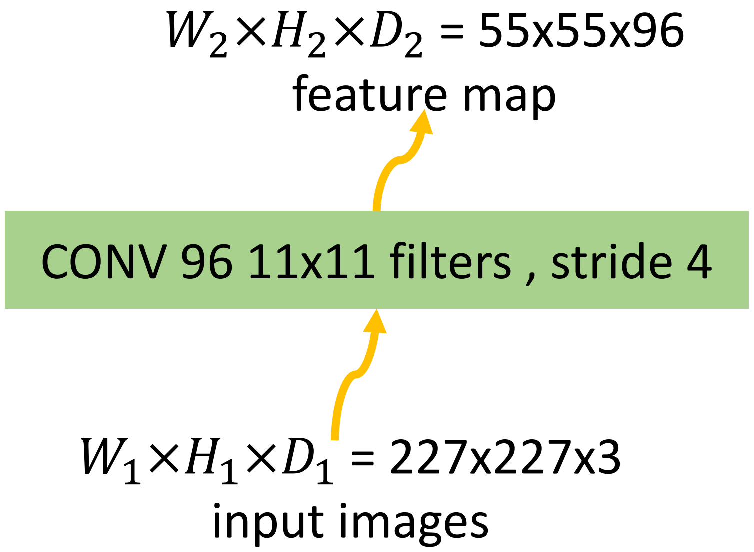
1D Pooling Layer
(1x2 filter, stride 2)





2D Pooling Layer
(2x2 filter, stride 2)

OUTPUT DIMENSIONS OF CONVOLUTION LAYERS



Input: accept a volume of size $W_1 \times H_1 \times D_1$

Hyperparameters:

(1) #filters: K (2) spatial extent: F

(3) stride: S (4) #paddings: P

Output: The number of parameters is $W_2 \times H_2 \times D_2$, where

- $W_2 = \frac{W_1 - F + 2P}{S} + 1$
- $H_2 = \frac{H_1 - F + 2P}{S} + 1$
- $D_2 = K$

ILLINOIS

QUIZ: NUMBER OF PARAMETERS + ReLU

$W_2 \times H_2 \times D_2$ = 55x55x96
feature map

CONV+ReLU 96 11x11 filters , stride 4

$W_1 \times H_1 \times D_1$ = 227x227x3
input images

Input: Width $W_1 = 227$; Height $H_1 = 227$; Depth $D_1 = 3$ (e.g., R,G,B channels)

Hyperparameters:
#filters: $K = 96$; spatial extent $F = 11$; stride $S = 4$; #paddings: $P = 0$

Question: What is the number of parameters?

Answer: $F \times F \times D_1 \times K + K = (11 \times 11) \times 3 \times 96 + 96$

OUTPUT DIMENSIONS OF POOLING LAYERS

$W_2 \times H_2 \times D_2$ = 27x27x96
feature map

↑

Max Pooling 3x3 filter, stride 2

↑

$W_1 \times H_1 \times D_1$ = 55x55x96
feature map

Input: accept a volume of size $W_1 \times H_1 \times D_1$

Hyperparameters:

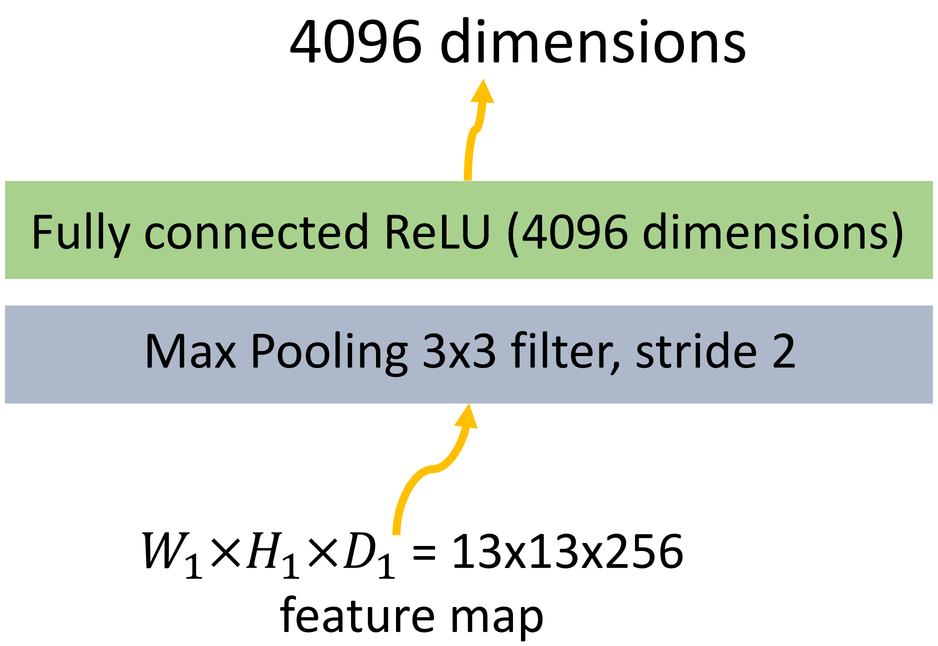
(1) spatial extent: F

(2) stride: S

Output: produce a volume of size $W_2 \times H_2 \times D_2$, where

- $W_2 = \frac{W_1 - F}{S} + 1$
- $H_2 = \frac{H_1 - F}{S} + 1$
- $D_2 = D_1$

QUIZ: NUMBER OF PARAMETERS OF FULLY CONNECTED LAYERS

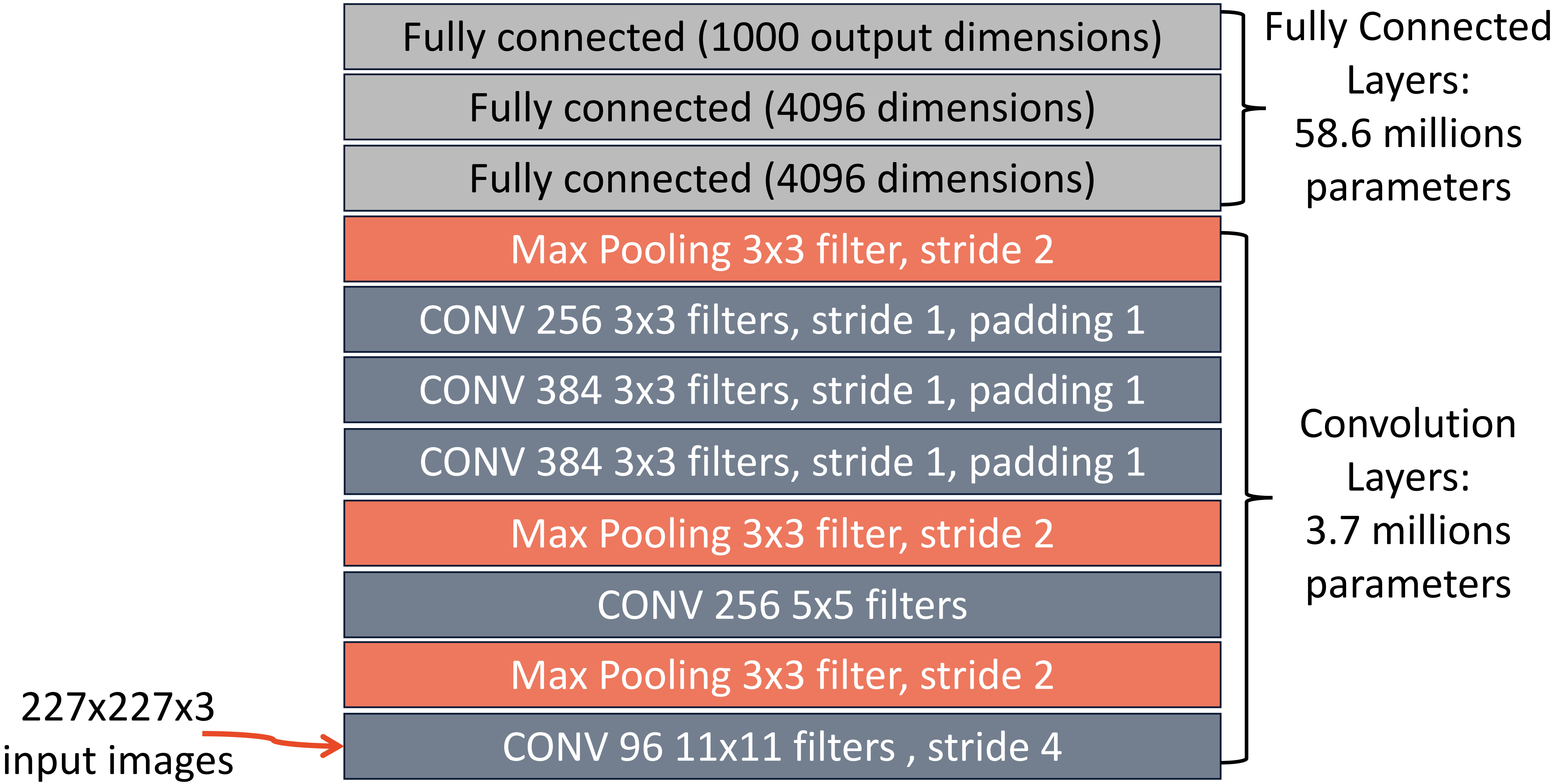


Input: volume of size $W_1 \times H_1 \times D_1$
Width $W_1 = 13$; Height $H_1 = 13$;
Depth $D_1 = 256$

Hyperparameters:
- Pooling: Spatial extent $F = 3$; stride $S = 2$
- Fully connected layer 4096 dimensions

Answer: Number of parameters:

6x6x256x4096 + 1


Fully connected (1000 output dimensions)
Fully connected (4096 dimensions)
Fully connected (4096 dimensions)
Max Pooling 3x3 filter, stride 2
CONV 256 3x3 filters, stride 1, padding 1
CONV 384 3x3 filters, stride 1, padding 1
CONV 384 3x3 filters, stride 1, padding 1
Max Pooling 3x3 filter, stride 2
CONV 256 5x5 filters
Max Pooling 3x3 filter, stride 2
CONV 96 11x11 filters , stride 4
227x227x3 input images
Fully Connected Layers: 58.6 millions parameters
Convolution Layers: 3.7 millions parameters

FORWARD CALCULATION OF CONVOLUTION LAYERS

$W_2 \times H_2 \times D_2$ = 55x55x96
feature map

CONV 96 11x11 filters , stride 4

$W_1 \times H_1 \times D_1$ = 227x227x3
input images

Input: accept a volume of size $W_1 \times H_1 \times D_1$

Hyperparameters:

(1) #filters: K (2) spatial extent: F

(3) stride: S (4) #paddings: P

The number of calculations:

$\underset{\textit{output size}}{(W_2 \times H_2 \times D_2)} \times \underset{\textit{filter size}}{(F \times F)} \times D_1$, where

- $W_2 = \frac{W_1 - F + 2P}{S} + 1$
- $H_2 = \frac{H_1 - F + 2P}{S} + 1$
- $D_2 = K$

ILLINOIS

FORWARD CALCULATION OF POOLING LAYERS

$W_2 \times H_2 \times D_2$ = 27x27x96
feature map

Max Pooling 3x3 filter, stride 2

$W_1 \times H_1 \times D_1$ = 55x55x96
feature map

Input: accept a volume of size
$W_1 \times H_1 \times D_1$

Hyperparameters:

(1) spatial extent: F

(3) stride: S

The number of calculations:

$$\underbrace{(W_2 \times H_2 \times D_2)}_{\textit{output size}} \times \underbrace{(F \times F)}_{\textit{filter size}}, \text{ where}$$

- $W_2 = \frac{W_1 - F}{S} + 1$
- $H_2 = \frac{H_1 - F}{S} + 1$
- $D_2 = D_1$

FORWARD CALCULATION OF FULLY-CONNECTED LAYERS

$4096\ dimensions$

Fully connected ReLU (4096 dimensions)

$W_1 \times H_1 \times D_1$ = 6x6x256
feature map

Input: accept a volume of size
$W_1 \times H_1 \times D_1$
$Width\ W_1 = 13;\ Height\ H_1 = 13;$
$Depth\ D_1 = 256$

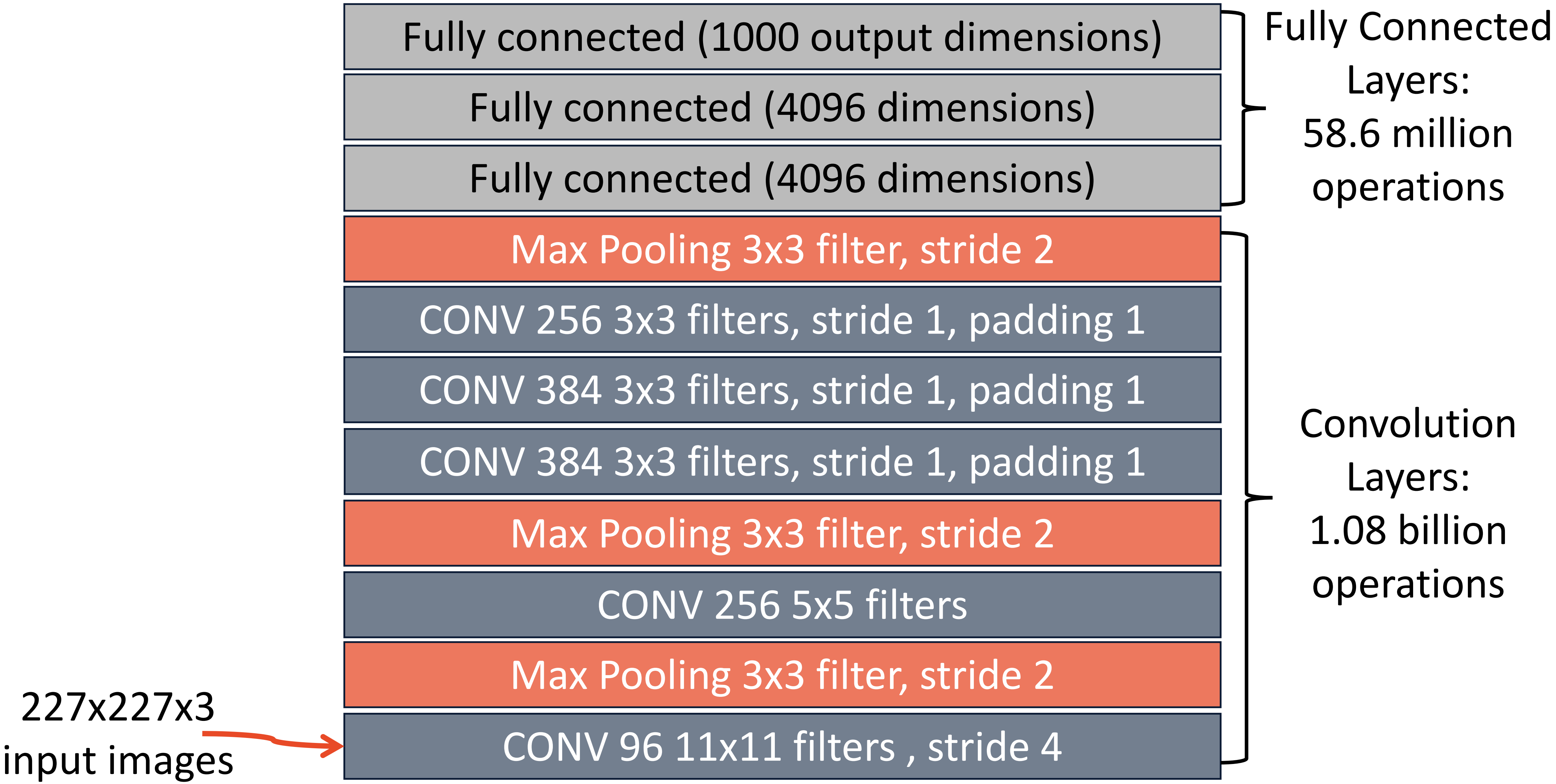
Hyperparameters:

- *Pooling: Spatial extent* $F = 3$; *stride* $S = 2$
- *Fully connected layer 4096 dimensions D*

The number of calculations:

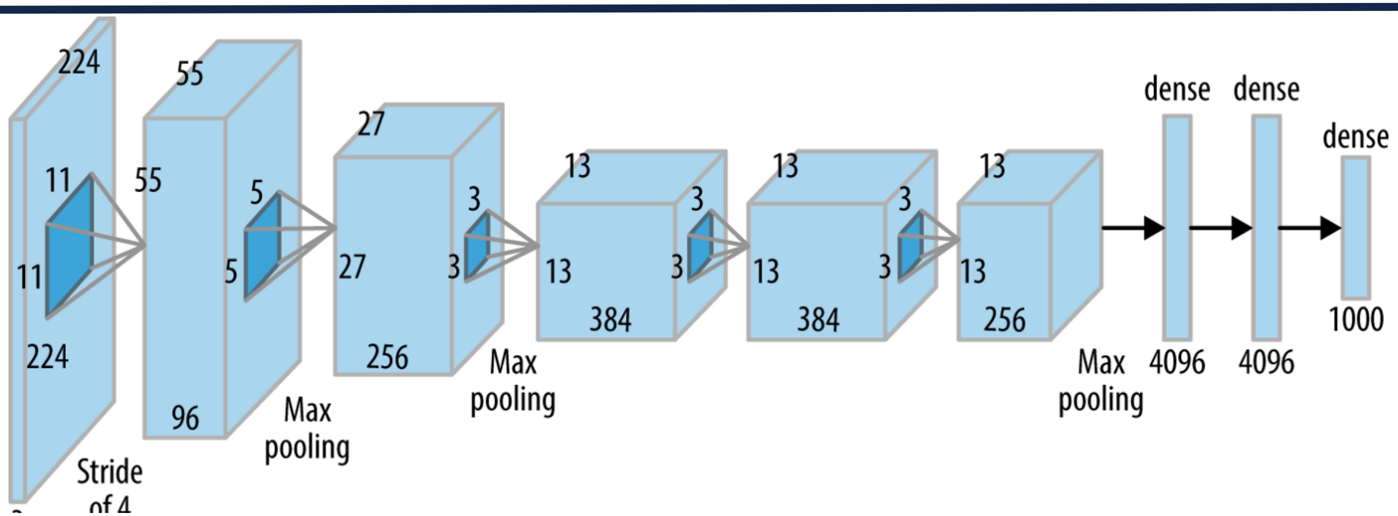
$$\underbrace{(W_1 \times H_1 \times D_1)}_{\textit{input size}} \times \underbrace{D}_{\textit{output size}}$$

ILLINOIS


Fully connected (1000 output dimensions)
Fully connected (4096 dimensions)
Fully connected (4096 dimensions)
Max Pooling 3x3 filter, stride 2
CONV 256 3x3 filters, stride 1, padding 1
CONV 384 3x3 filters, stride 1, padding 1
CONV 384 3x3 filters, stride 1, padding 1
Max Pooling 3x3 filter, stride 2
CONV 256 5x5 filters
Max Pooling 3x3 filter, stride 2
CONV 96 11x11 filters , stride 4
Fully Connected Layers: 58.6 million operations
Convolution Layers: 1.08 billion operations
227x227x3 input images

CNN ARCHITECTURES

AlexNet



AlexNet Network - Structural Details													
Input			Output			Layer	Stride	Pad	Kernel size		in	out	# of Param
227	227	3	55	55	96	conv1	4	0	11	11	3	96	34944
55	55	96	27	27	96	maxpool1	2	0	3	3	96	96	0
27	27	96	27	27	256	conv2	1	2	5	5	96	256	614656
27	27	256	13	13	256	maxpool2	2	0	3	3	256	256	0
13	13	256	13	13	384	conv3	1	1	3	3	256	384	885120
13	13	384	13	13	384	conv4	1	1	3	3	384	384	1327488
13	13	384	13	13	256	conv5	1	1	3	3	384	256	884992
13	13	256	6	6	256	maxpool5	2	0	3	3	256	256	0
						fc6			1	1	9216	4096	37752832
						fc7			1	1	4096	4096	16781312
						fc8			1	1	4096	1000	4097000
Total													62,378,344

AlexNet: Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks.", NIPS 2012

AlexNet

Fully connected (1000 output dimensions)

Fully connected (4096 dimensions)

Fully connected (4096 dimensions)

Max Pooling 3x3 filter, stride 2

CONV 256 3x3 filters, stride 1, padding 1

CONV 384 3x3 filters, stride 1, padding 1

CONV 384 3x3 filters, stride 1, padding 1

Max Pooling 3x3 filter, stride 2

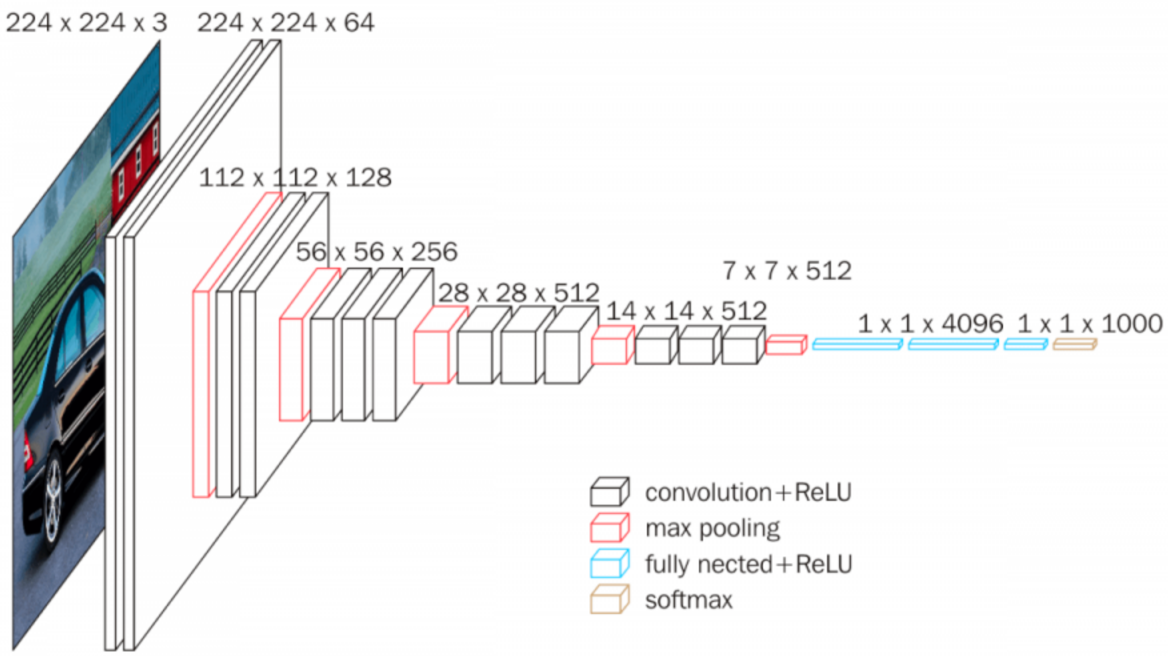
CONV 256 5x5 filters

Max Pooling 3x3 filter, stride 2

CONV 96 11x11 filters , stride 4

227x227x3
input images

VGG



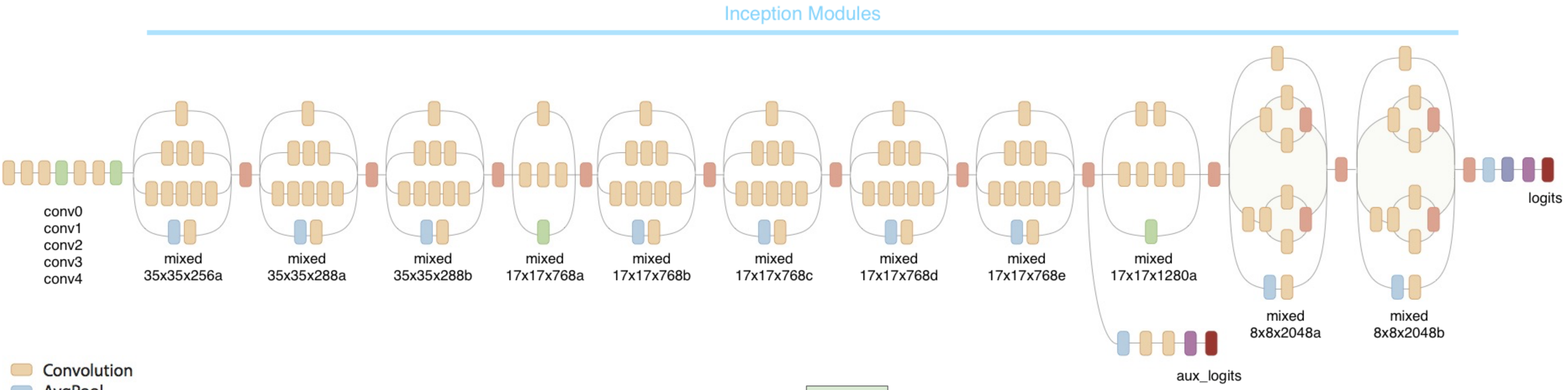
VGG16 - Structural Details													
#	Input Image			output			Layer	Stride	Kernel		in	out	Param
1	224	224	3	224	224	64	conv3-64	1	3	3	3	64	1792
2	224	224	64	224	224	64	conv3064	1	3	3	64	64	36928
	224	224	64	112	112	64	maxpool	2	2	2	64	64	0
3	112	112	64	112	112	128	conv3-128	1	3	3	64	128	73856
4	112	112	128	112	112	128	conv3-128	1	3	3	128	128	147584
	112	112	128	56	56	128	maxpool	2	2	2	128	128	65664
5	56	56	128	56	56	256	conv3-256	1	3	3	128	256	295168
6	56	56	256	56	56	256	conv3-256	1	3	3	256	256	590080
7	56	56	256	56	56	256	conv3-256	1	3	3	256	256	590080
	56	56	256	28	28	256	maxpool	2	2	2	256	256	0
8	28	28	256	28	28	512	conv3-512	1	3	3	256	512	1180160
9	28	28	512	28	28	512	conv3-512	1	3	3	512	512	2359808
10	28	28	512	28	28	512	conv3-512	1	3	3	512	512	2359808
	28	28	512	14	14	512	maxpool	2	2	2	512	512	0
11	14	14	512	14	14	512	conv3-512	1	3	3	512	512	2359808
12	14	14	512	14	14	512	conv3-512	1	3	3	512	512	2359808
13	14	14	512	14	14	512	conv3-512	1	3	3	512	512	2359808
	14	14	512	7	7	512	maxpool	2	2	2	512	512	0
14	1	1	25088	1	1	4096	fc		1	1	25088	4096	102764544
15	1	1	4096	1	1	4096	fc		1	1	4096	4096	16781312
16	1	1	4096	1	1	1000	fc		1	1	4096	1000	4097000
							Total						138,423,208

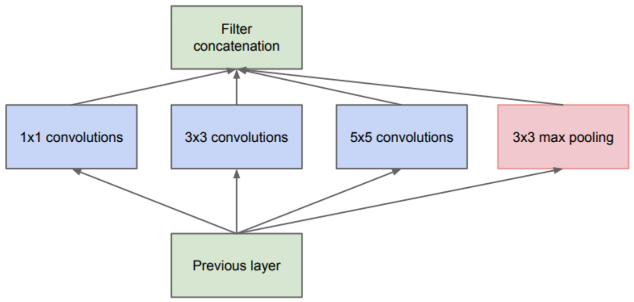
Deeper network than AlexNet, smaller filter size

VGGNet: Karen Simonyan, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition.", ICLR 2015

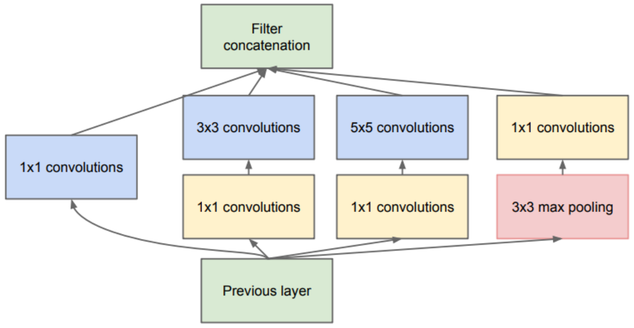
ILLINOIS

INCEPTION





(a) Inception module, naïve version



(b) Inception module with dimensionality reduction

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going deeper with convolutions.", CVPR 2015

ILLINOIS

INCEPTION

GoogLeNet - Structural Details

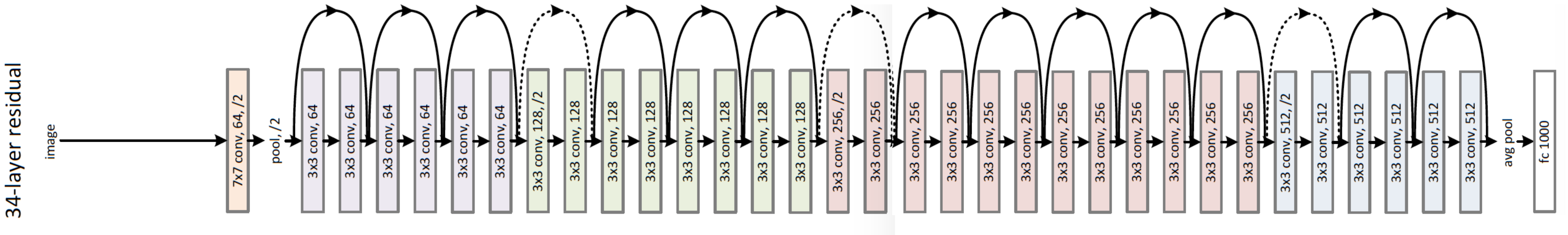
	Input Image			output			Layer	Input Layer	Stride	Pad	Kernel		in	out	Param
	227	227	3	112	112	64	conv1	input	2	1	7	7	3	64	9472
	112	112	64	56	56	64	maxpool1	conv1	2	0.5	3	3	64	64	0
	56	56	64	56	56	64	conv1x1	maxpool1	1	0	1	1	64	64	4160
	56	56	64	56	56	192	conv2-1		1	1	3	3	64	192	110784
	56	56	192	28	28	192	maxpool2		2	0.5	3	3	192	192	0
inception (3a)	28	28	192	28	28	96	conv1x1a	maxpool2	1	0	1	1	192	96	18528
	28	28	96	28	28	16	conv1x1b	maxpool2	1	0	1	1	192	16	3088
	28	28	192	28	28	192	maxpool-a	maxpool2	1	1	3	3	192	192	0
	28	28	192	28	28	64	conv1x1c	maxpool2	1	0	1	1	192	64	12352
	28	28	96	28	28	128	conv3-3	conv1x1a	1	1	3	3	96	128	110720
	28	28	16	28	28	32	conv5x5	conv1x1b	1	2	5	5	16	32	12832
	28	28	192	28	28	32	conv1x1d	maxpool-a	1	0	1	1	192	32	6176
				28	28	256	depth-concat	conv1x1c, conv3x3, conv5x5, conv1x1d							
inception (3b)	28	28	256	28	28	128	conv1x1a	depth-concat	1	0	1	1	256	128	32896
	28	28	128	28	28	32	conv1x1b	depth-concat	1	0	1	1	256	32	8224
	28	28	192	28	28	256	maxpool-a	depth-concat	1	1	3	3	256	256	0
	28	28	192	28	28	128	conv1x1c	depth-concat	1	0	1	1	256	128	32896
	28	28	96	28	28	192	conv3-3	conv1x1a	1	1	3	3	128	192	221376
	28	28	16	28	28	96	conv5x5	conv1x1b	1	2	5	5	32	96	76896
	28	28	192	28	28	64	conv1x1d	maxpool-a	1	0	1	1	256	64	16448
				28	28	480	depth-concat	conv1x1c, conv3x3, conv5x5, conv1x1d							
	28	28	480	14	14	480	maxpool3	depth-concat	2	0.5	3	3	480	480	0
inception (4a)	14	14	480	14	14	96	conv1x1a	maxpool3	1	0	1	1	480	96	46176
	14	14	480	14	14	16	conv1x1b	maxpool3	1	0	1	1	480	16	7696
	14	14	480	14	14	480	maxpool-a	maxpool3	1	1	3	3	480	480	0
	14	14	480	14	14	192	conv1x1c	maxpool3	1	0	1	1	480	192	92352
	14	14	96	14	14	208	conv3-3	conv1x1a	1	1	3	3	96	208	179920
	14	14	16	14	14	48	conv5x5	conv1x1b	1	2	5	5	16	48	19248
	14	14	192	14	14	64	conv1x1d	maxpool-a	1	0	1	1	480	64	30784
				14	14	512	depth-concat	conv1x1c, conv3x3, conv5x5, conv1x1d							
inception (4b)	14	14	512	14	14	112	conv1x1a	depth-concat	1	0	1	1	512	112	57456
	14	14	512	14	14	24	conv1x1b	depth-concat	1	0	1	1	64	24	1560
	14	14	512	14	14	64	maxpool-a	depth-concat	1	1	3	3	64	64	0
	14	14	512	14	14	160	conv1x1c	depth-concat	1	0	1	1	64	160	10400
	14	14	96	14	14	224	conv3-3	conv1x1a	1	1	3	3	112	224	226016
	14	14	16	14	14	64	conv5x5	conv1x1b	1	2	5	5	24	64	38464
	14	14	160	14	14	64	conv1x1d	maxpool-a	1	0	1	1	64	64	4160
				14	14	512	depth-concat	conv1x1c, conv3x3, conv5x5, conv1x1d							
inception (4c)	14	14	512	14	14	128	conv1x1a	depth-concat	1	0	1	1	512	128	65664
	14	14	512	14	14	24	conv1x1b	depth-concat	1	0	1	1	64	24	1560
	14	14	512	14	14	64	maxpool-a	depth-concat	1	1	3	3	64	64	0
	14	14	512	14	14	128	conv1x1c	depth-concat	1	0	1	1	64	128	8320
	14	14	96	14	14	256	conv3-3	conv1x1a	1	1	3	3	128	256	295168
	14	14	16	14	14	64	conv5x5	conv1x1b	1	2	5	5	24	64	38464
	14	14	128	14	14	64	conv1x1d	maxpool-a	1	0	1	1	64	64	4160
				14	14	512	depth-concat	conv1x1c, conv3x3, conv5x5, conv1x1d							
inception (4d)	14	14	512	14	14	144	conv1x1a	depth-concat	1	0	1	1	512	144	73872
	14	14	512	14	14	32	conv1x1b	depth-concat	1	0	1	1	64	32	2080
	14	14	512	14	14	64	maxpool-a	depth-concat	1	1	3	3	64	64	0
	14	14	512	14	14	112	conv1x1c	depth-concat	1	0	1	1	64	112	7280
	14	14	96	14	14	288	conv3-3	conv1x1a	1	1	3	3	144	288	373536
	14	14	16	14	14	64	conv5x5	conv1x1b	1	2	5	5	32	64	51264
	14	14	112	14	14	64	conv1x1d	maxpool-a	1	0	1	1	64	64	4160
				14	14	528	depth-concat	conv1x1c, conv3x3, conv5x5, conv1x1d							
inception (4e)	14	14	528	14	14	160	conv1x1a	depth-concat	1	0	1	1	528	160	84640
	14	14	528	14	14	32	conv1x1b	depth-concat	1	0	1	1	64	32	2080
	14	14	528	14	14	64	maxpool-a	depth-concat	1	1	3	3	64	64	0
	14	14	528	14	14	256	conv1x1c	depth-concat	1	0	1	1	64	256	16640
	14	14	96	14	14	320	conv3-3	conv1x1a	1	1	3	3	160	320	461120
	14	14	16	14	14	128	conv5x5	conv1x1b	1	2	5	5	32	128	102528
	14	14	256	14	14	128	conv1x1d	maxpool-a	1	0	1	1	64	128	8320
				14	14	832	depth-concat	conv1x1c, conv3x3, conv5x5, conv1x1d							
	14	14	832	7	7	832	maxpool4	depth-concat	2	0.5	3	3	832	832	0
inception (5a)	7	7	832	7	7	160	conv1x1a	maxpool4	1	0	1	1	832	160	133280
	7	7	832	7	7	32	conv1x1b	maxpool4	1	0	1	1	832	32	26656
	7	7	832	7	7	832	maxpool-a	maxpool4	1	1	3	3	832	832	0
	7	7	832	7	7	256	conv1x1c	maxpool4	1	0	1	1	832	256	213248
	7	7	96	7	7	320	conv3-3	conv1x1a	1	1	3	3	160	320	461120
	7	7	16	7	7	128	conv5x5	conv1x1b	1	2	5	5	32	128	102528
	7	7	256	7	7	128	conv1x1d	maxpool-a	1	0	1	1	832	128	106624
				7	7	832	depth-concat	conv1x1c, conv3x3, conv5x5, conv1x1d							
inception (5b)	7	7	832	7	7	192	conv1x1a	depth-concat	1	0	1	1	832	192	159936
	7	7	832	7	7	48	conv1x1b	depth-concat	1	0	1	1	832	48	39984
	7	7	832	7	7	832	maxpool-a	depth-concat	1	1	3	3	832	832	0
	7	7	832	7	7	384	conv1x1c	depth-concat	1	0	1	1	832	384	319872
	7	7	96	7	7	384	conv3-3	conv1x1a	1	1	3	3	192	384	663936
	7	7	16	7	7	128	conv5x5	conv1x1b	1	2	5	5	48	128	153728
	7	7	384	7	7	128	conv1x1d	maxpool-a	1	0	1	1	128	128	16512
				7	7	1024	depth-concat	conv1x1c, conv3x3, conv5x5, conv1x1d							
	7	7	1024	1	1	1024	avgpool	depth-concat	1	0	7	7	1024	1024	0
	1	1	1024	1	1	1000	fc	depth-concat	1	0	1	1	1024	1000	1025000
									Total						6,414,360

ILLINOIS

ResNet

THANKS TO SHORTCUT CONNECTIONS:

- alleviate vanishing gradient problem
- combine shallow and deep networks

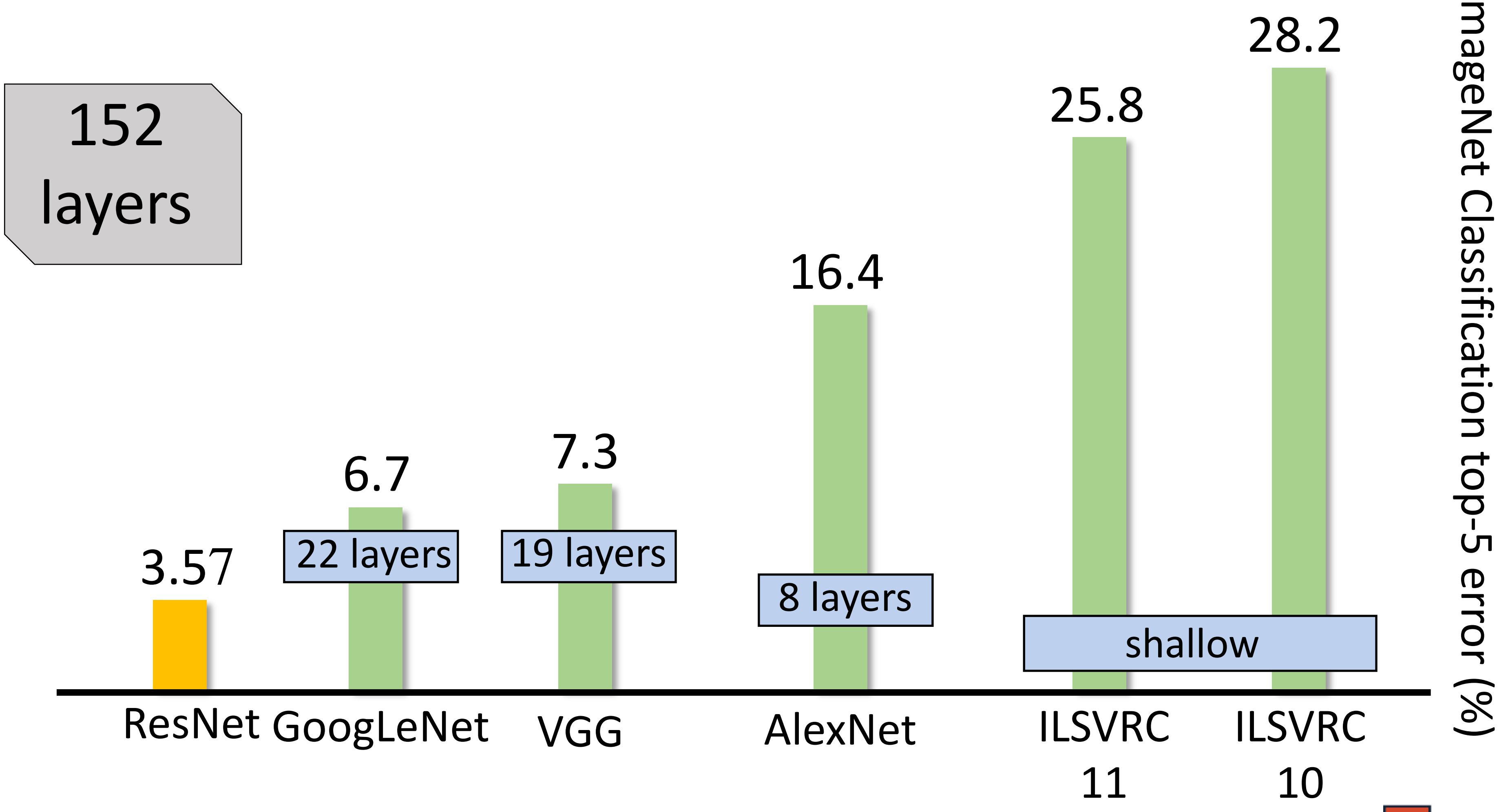


Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition", CVPR 2016

ResNet

ResNet18 - Structural Details														
#	Input Image			output			Layer	Stride	Pad	Kernel		in	out	Param
1	227	227	3	112	112	64	conv1	2	1	7	7	3	64	9472
	112	112	64	56	56	64	maxpool	2	0.5	3	3	64	64	0
2	56	56	64	56	56	64	conv2-1	1	1	3	3	64	64	36928
3	56	56	64	56	56	64	conv2-2	1	1	3	3	64	64	36928
4	56	56	64	56	56	64	conv2-3	1	1	3	3	64	64	36928
5	56	56	64	56	56	64	conv2-4	1	1	3	3	64	64	36928
6	56	56	64	28	28	128	conv3-1	2	0.5	3	3	64	128	73856
7	28	28	128	28	28	128	conv3-2	1	1	3	3	128	128	147584
8	28	28	128	28	28	128	conv3-3	1	1	3	3	128	128	147584
9	28	28	128	28	28	128	conv3-4	1	1	3	3	128	128	147584
10	28	28	128	14	14	256	conv4-1	2	0.5	3	3	128	256	295168
11	14	14	256	14	14	256	conv4-2	1	1	3	3	256	256	590080
12	14	14	256	14	14	256	conv4-3	1	1	3	3	256	256	590080
13	14	14	256	14	14	256	conv4-4	1	1	3	3	256	256	590080
14	14	14	256	7	7	512	conv5-1	2	0.5	3	3	256	512	1180160
15	7	7	512	7	7	512	conv5-2	1	1	3	3	512	512	2359808
16	7	7	512	7	7	512	conv5-3	1	1	3	3	512	512	2359808
17	7	7	512	7	7	512	conv5-4	1	1	3	3	512	512	2359808
	7	7	512	1	1	512	avg pool	7	0	7	7	512	512	0
18	1	1	512	1	1	1000	fc					512	1000	513000
Total														**11,511,784**

EVOLUTION OF CNN ARCHITECTURES



ILLINOIS

EVOLUTION OF CNN ARCHITECTURES

Comparison					
Network	Year	Salient Feature	top5 accuracy	Parameters	FLOP
AlexNet	2012	Deeper	84.70%	62M	1.5B
VGGNet	2014	Fixed-size kernels	92.30%	138M	19.6B
Inception	2014	Wider - Parallel kernels	93.30%	6.4M	2B
ResNet-152	2015	Shortcut connections	95.51%	60.3M	11B

HEALTHCARE APPLICATIONS OF CNN

Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs

Gulshan, Varun, Lily Peng, Marc Coram, Martin C. Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, et al. 2016. “Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs.” *JAMA: The Journal of the American Medical Association* 316 (22): 2402–10.

ILLINOIS

DIABETIC RETINOPATHY DIAGNOSIS

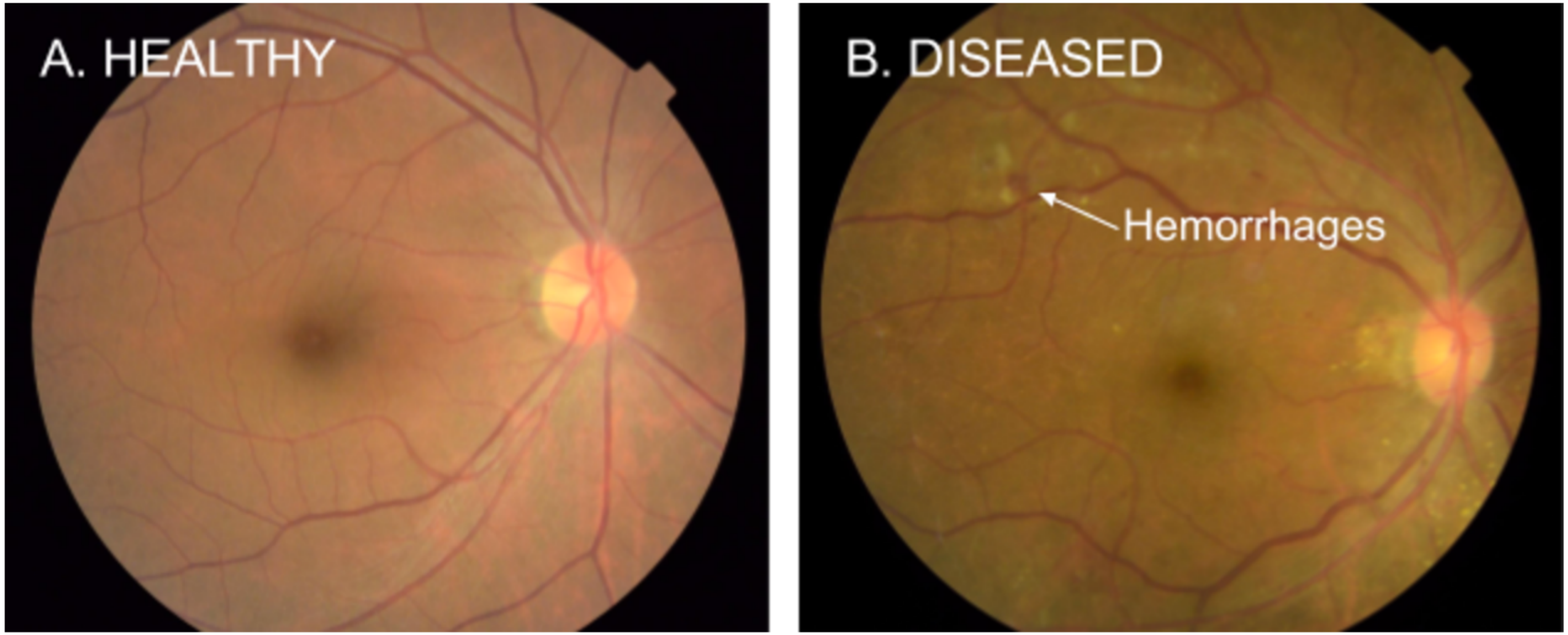
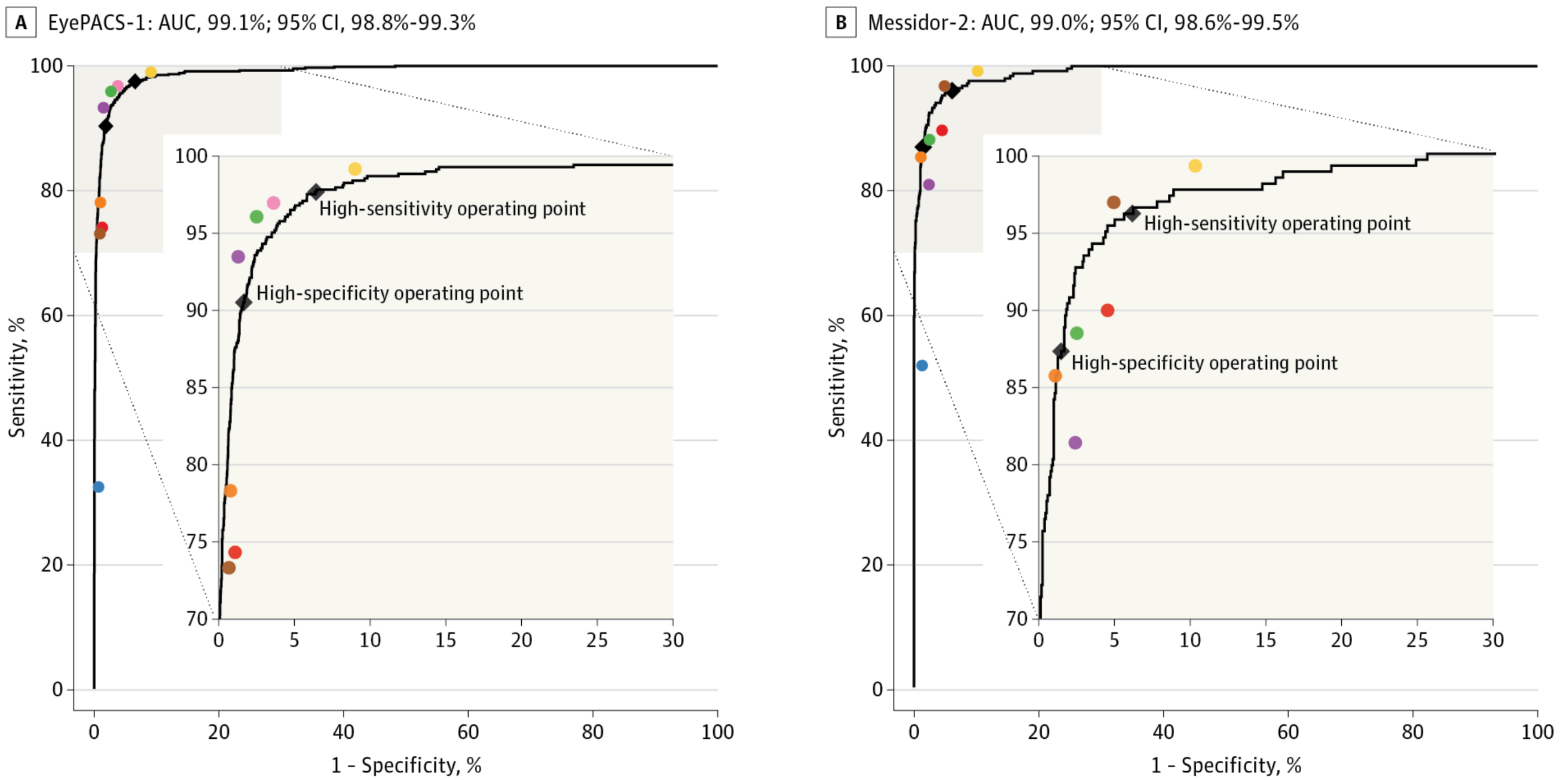


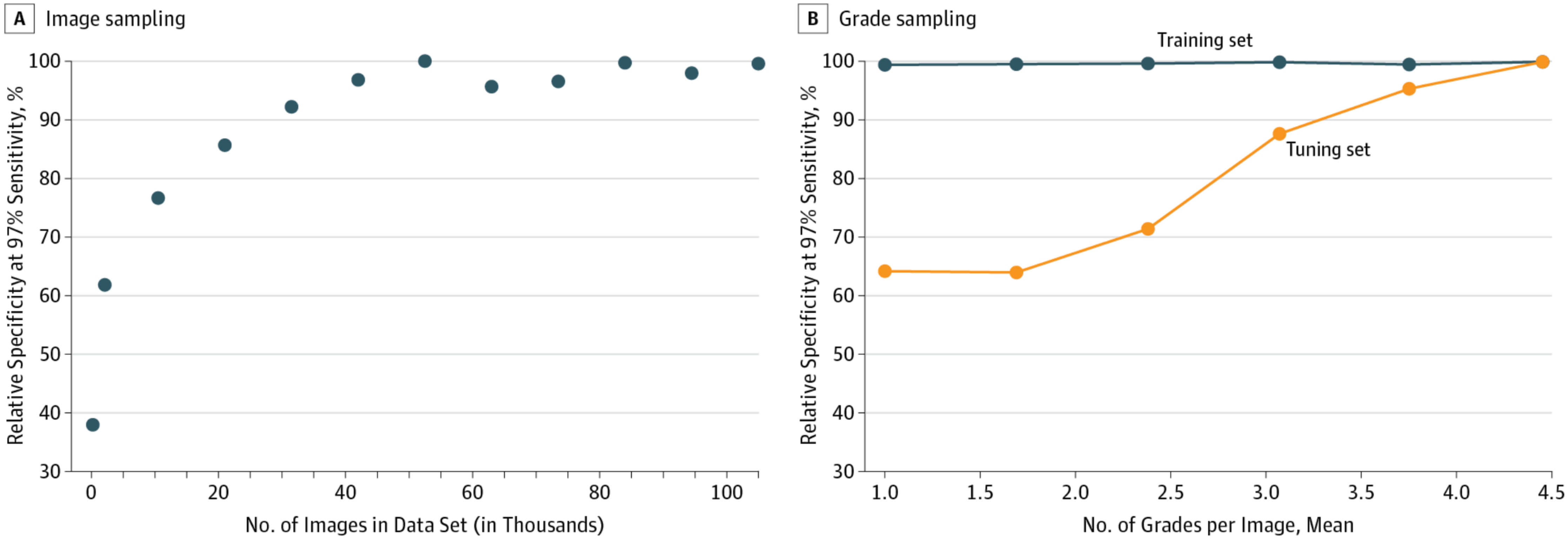
Figure 1. Examples of retinal fundus photographs that are taken to screen for DR. The image on the left is of a healthy retina (A), whereas the image on the right is a retina with referable diabetic retinopathy (B) due a number of hemorrhages (red spots) present.

DIABETIC RETINOPATHY DIAGNOSIS



ILLINOIS

DIABETIC RETINOPATHY DIAGNOSIS



ILLINOIS

Dermatologist-level classification of skin cancer with deep neural networks

A. Esteva, B. Kuprel, R.A. Novoa, J. Ko, S.M. Swetter, H.M. Blau , S. Thrun. Dermatologist-level classification of skin cancer with deep neural networks. Nature 542, 115–118 (2017)

ILLINOIS

DETECTING SKIN CANCER

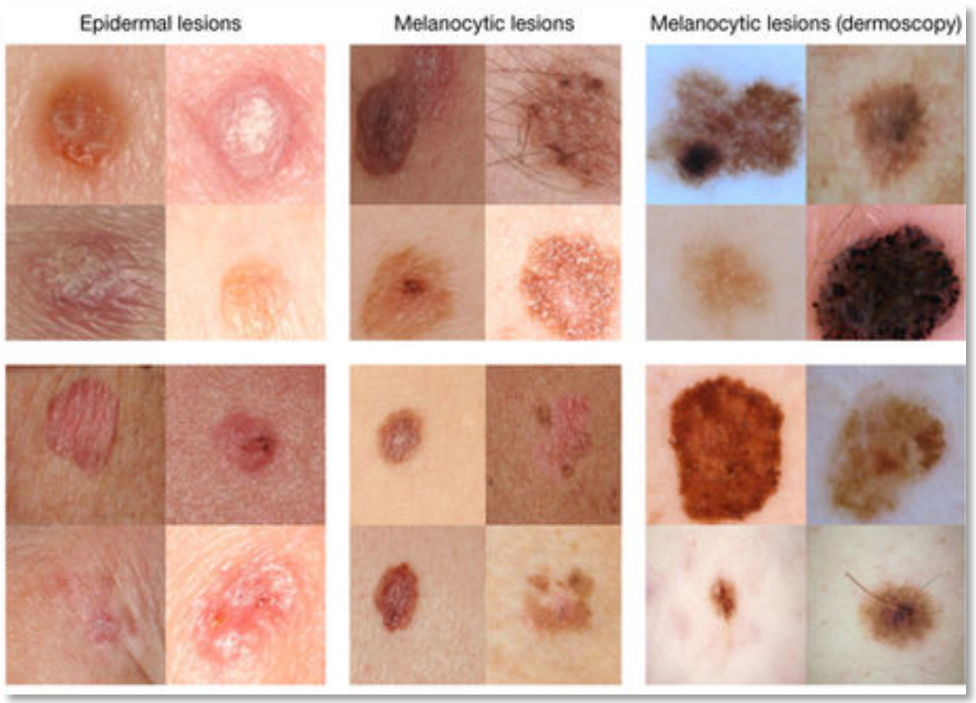
GIVEN CLINICAL IMAGES, CLASSIFY:

- Keratinocyte carcinomas VS benign seborrheic keratoses
- malignant melanomas VS benign nevi

Benign

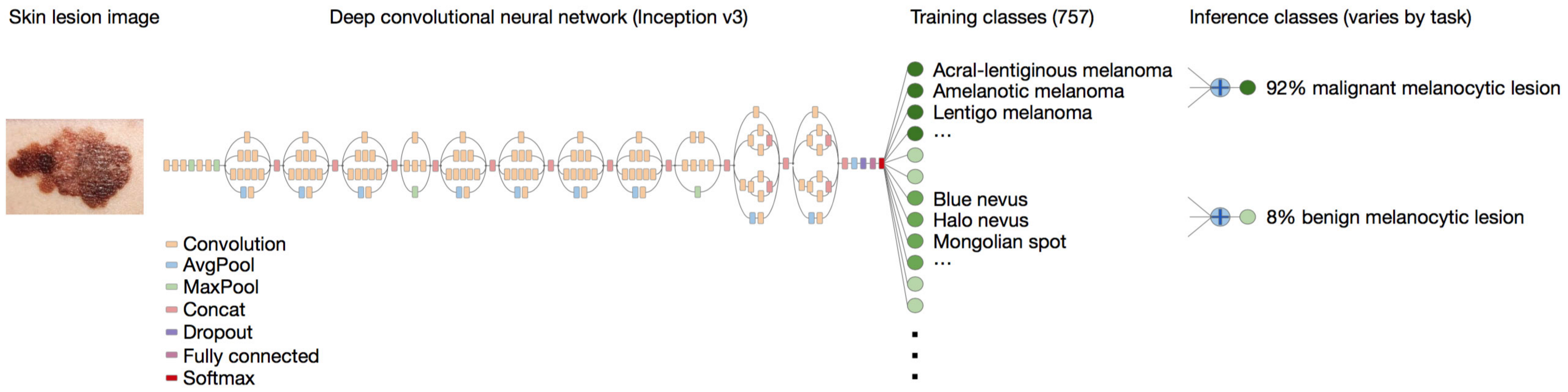
Benign or Malignant?

Malignant

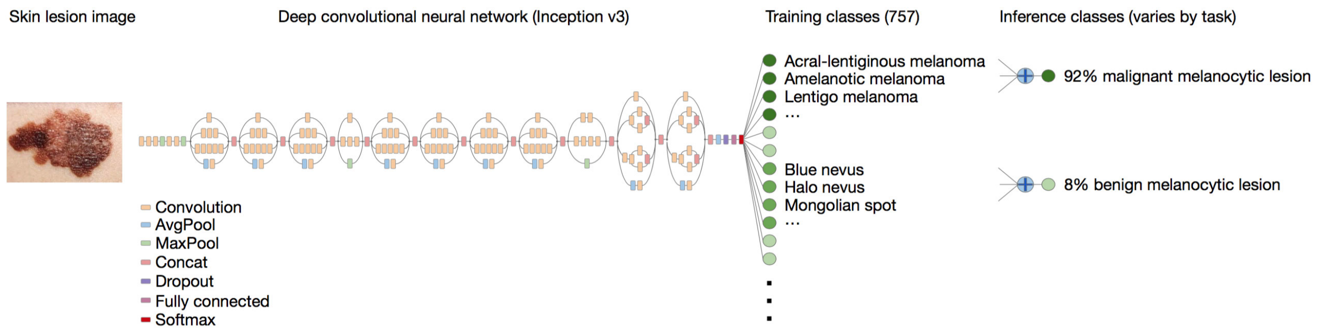


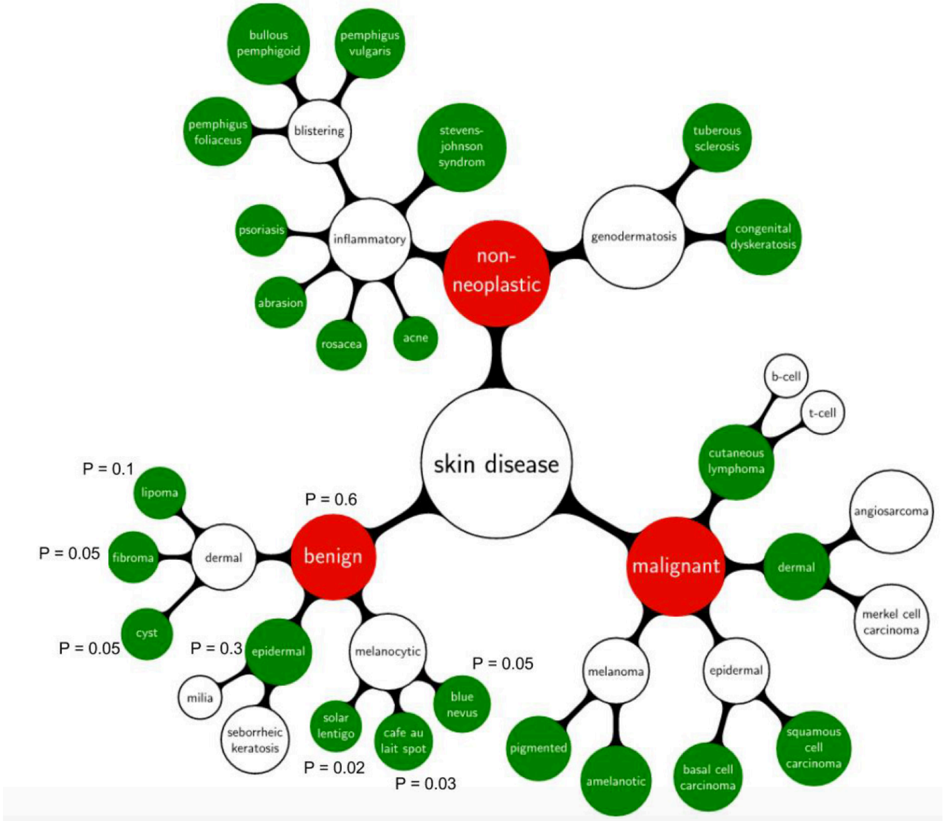
ILLINOIS

MODEL ARCHITECTURE: INCEPTION V3



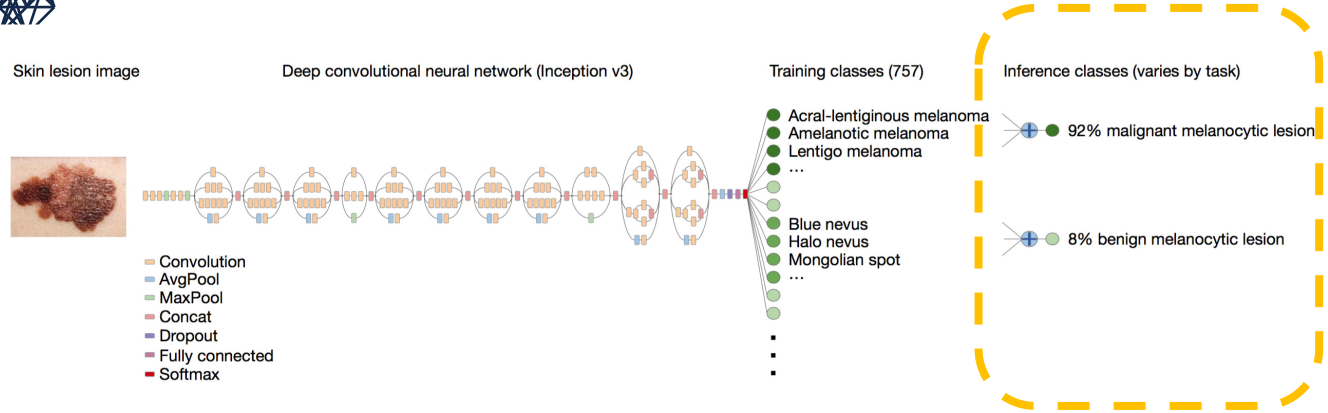
MODEL ARCHITECTURE: INCEPTION V3

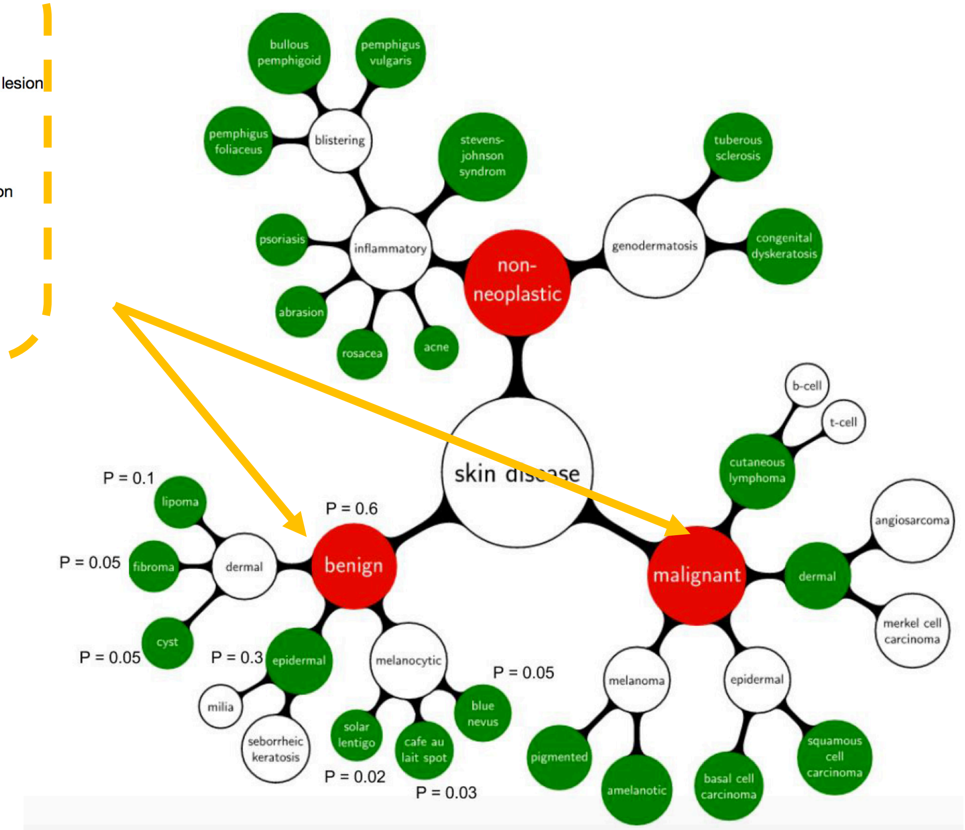




Disease hierarchy curated by experts

MODEL ARCHITECTURE: INCEPTION V3



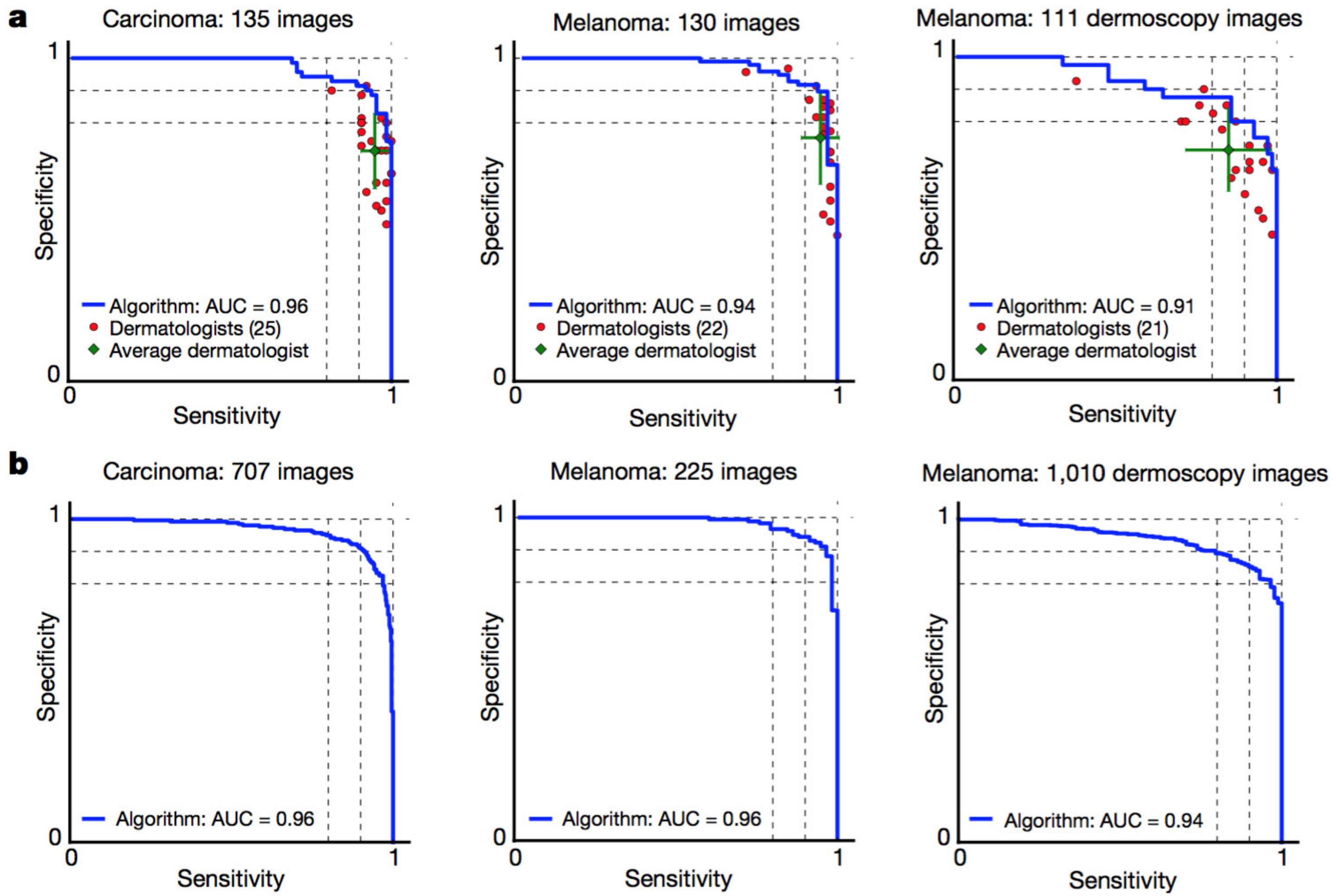


Disease hierarchy curated by experts

ILLINOIS

MODEL PERFORMANCE

- Better than average board-certified dermatologists
- More performance improvement given more data



ILLINOIS