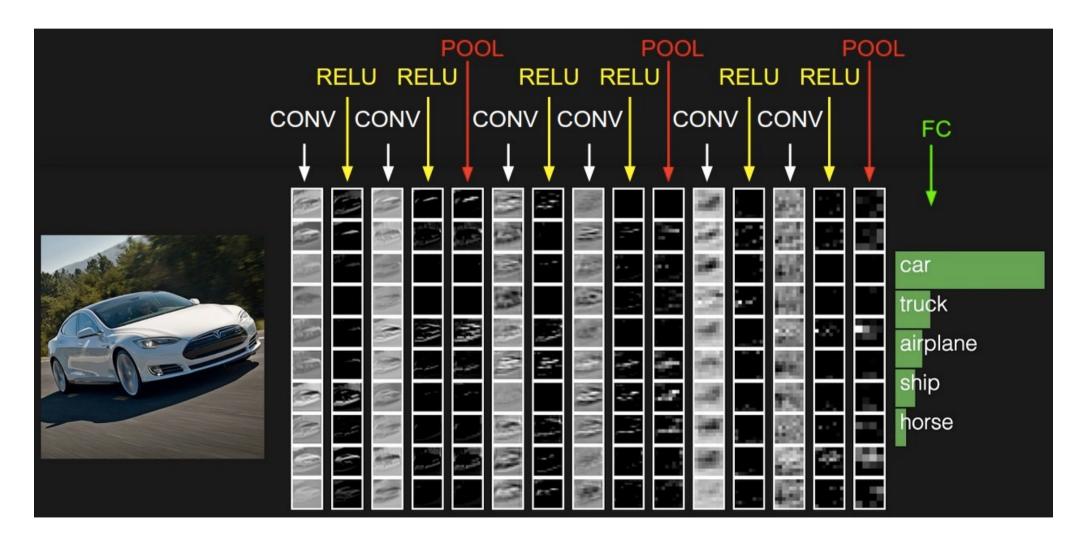
Convolutional neural networks

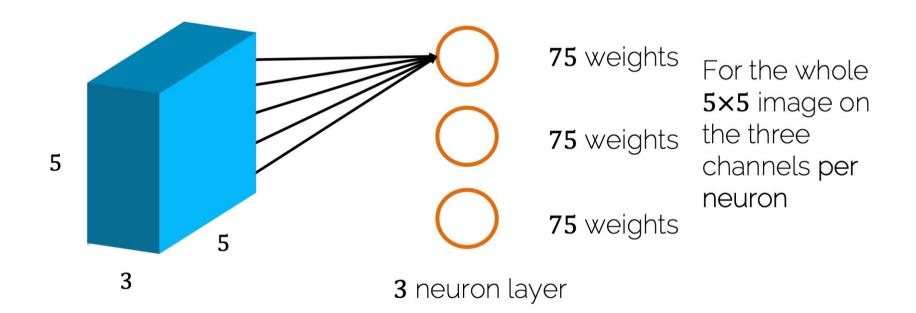


Many slides from Rob Fergus, Andrej Karpathy

Outline

- Building blocks
- Motivation and history
- ImageNet challenge
- Architectures:
 - 1st generation (2012-2013): AlexNet
 - 2nd generation (2014): VGGNet, GoogLeNet
 - 3rd generation (2015): ResNet
 - 4th generation (2016): ResNeXt, DenseNet

Fully connected NN for images

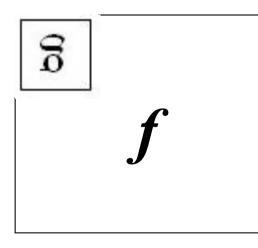


Impractical Does not scale well Does not exploit the spatial structure of images

Convolution - Image Filtering

Let f be the image and g be the kernel. The output of convolving f with g is denoted f * g.

$$(f * g)[m, n] = \sum_{k, l} f[m - k, n - l]g[k, l]$$



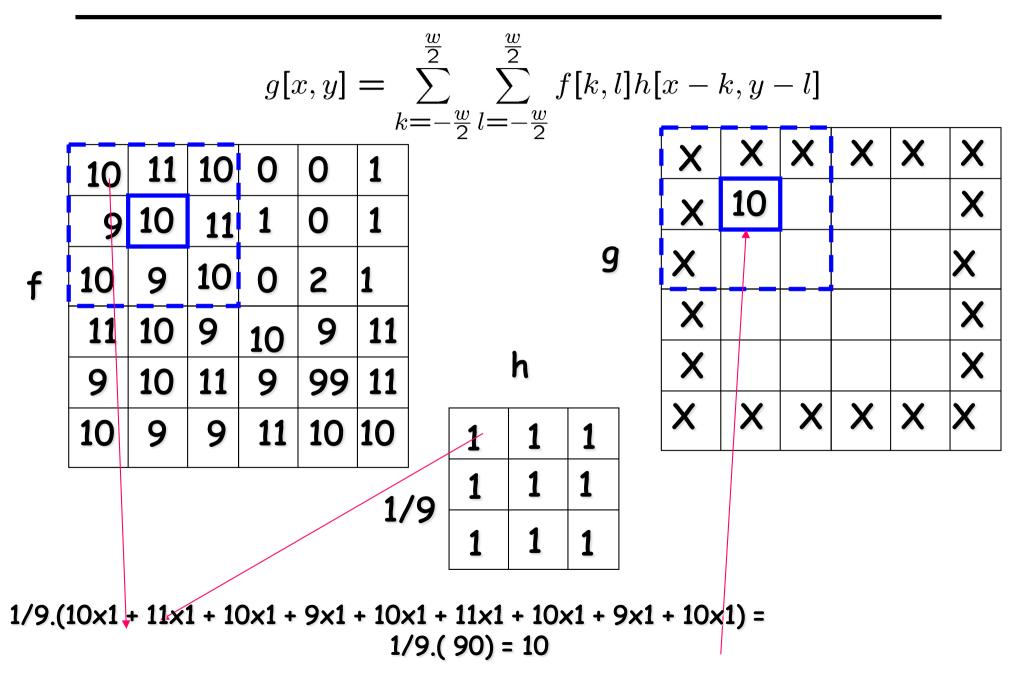
Convention: kernel is "flipped"

In words: value of the filtered image at particular location:

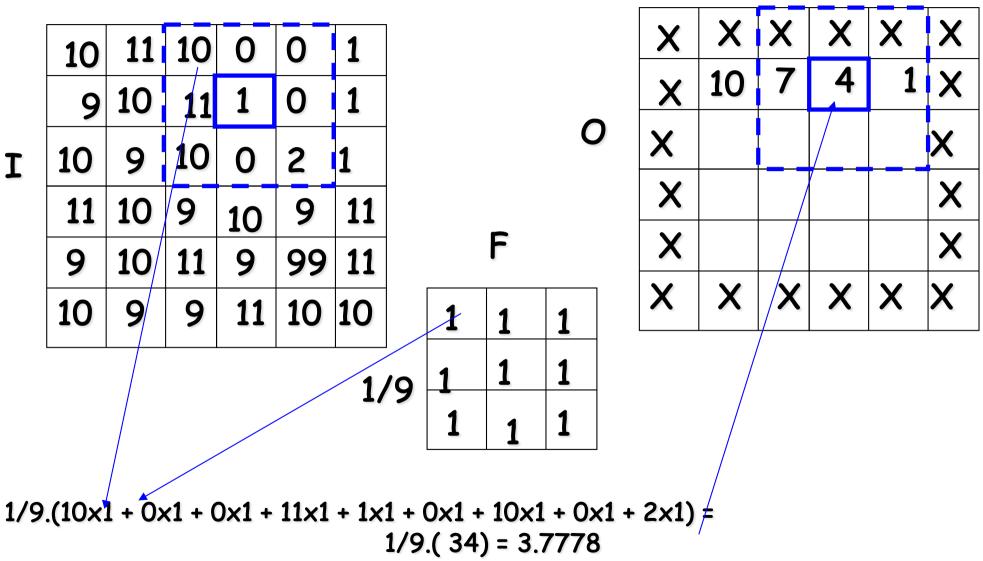
- 1. Overlay filter over the image centered at that location,
- 2. Multiply the image values by filter coefficients and sum the result
- 3. To produce the filtered image slide the filter over every location and perform this operation

Source: F. Durand

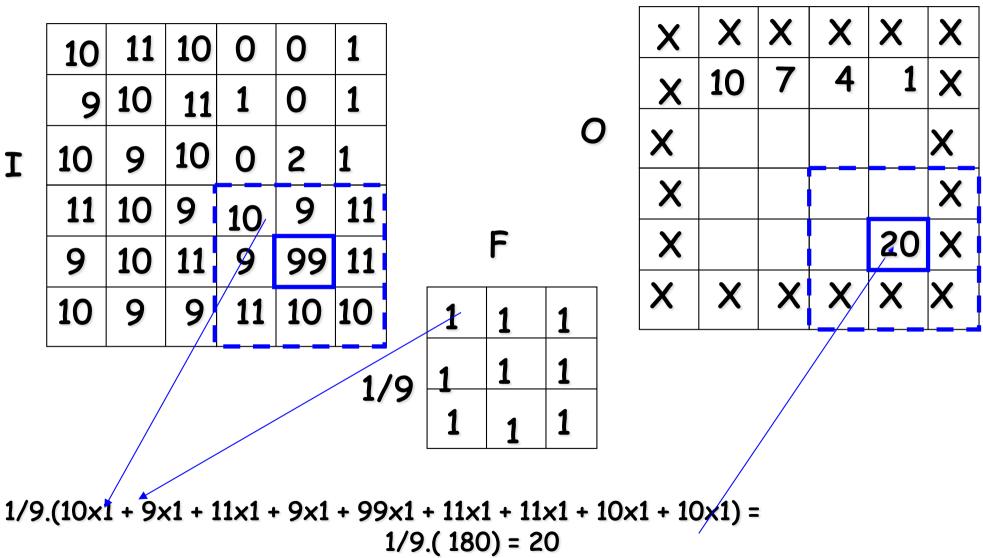
Convolution in 2D



Example:

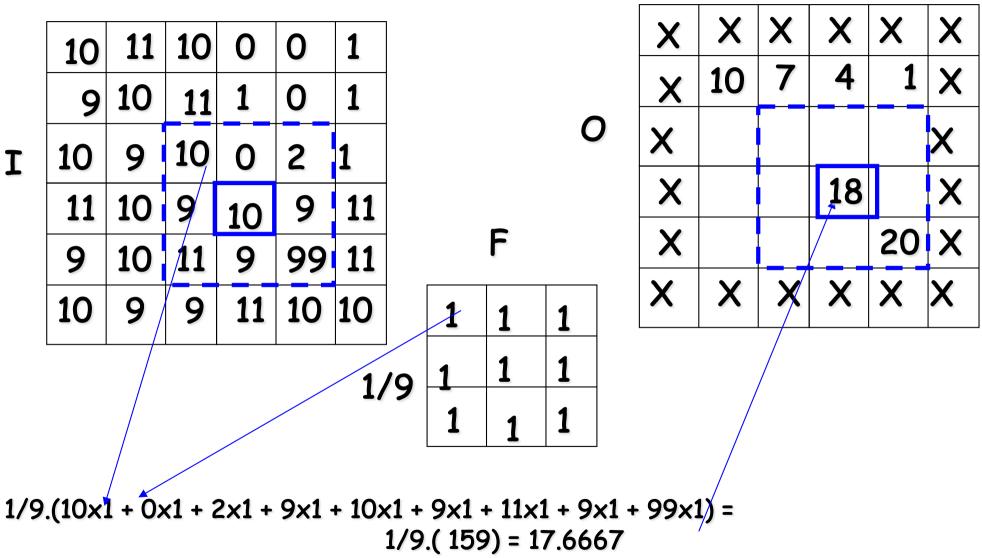


Example:



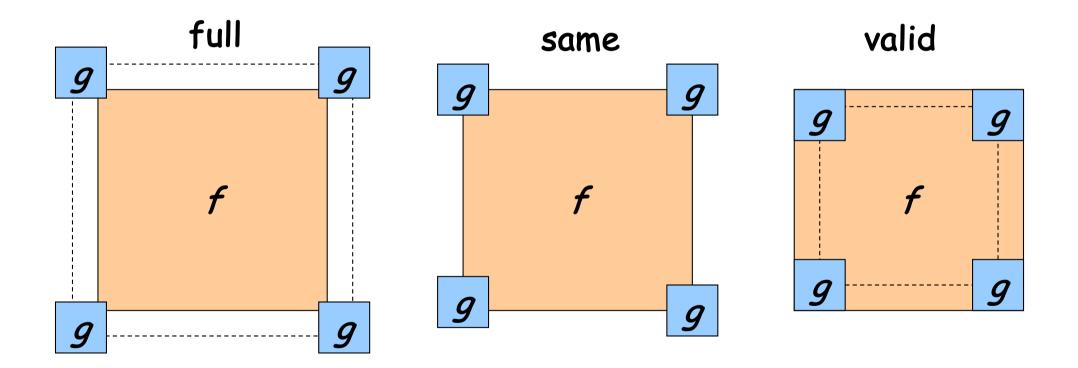
Ι

Example:

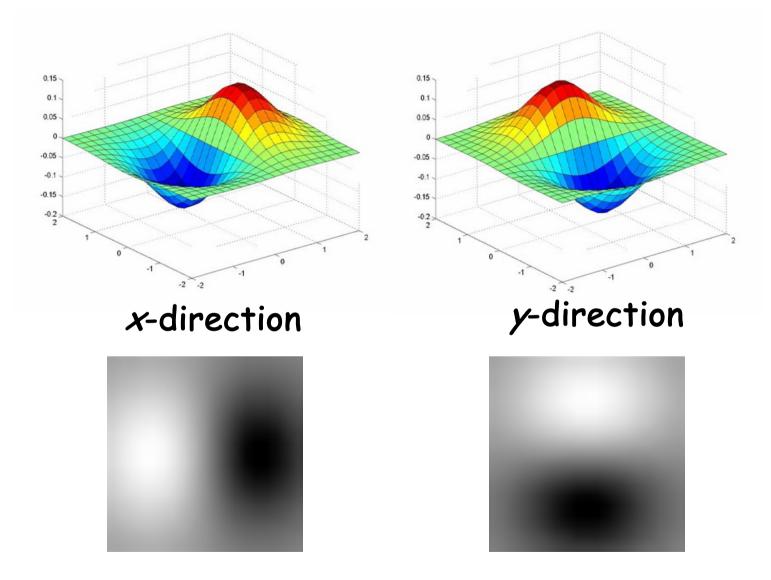


Details

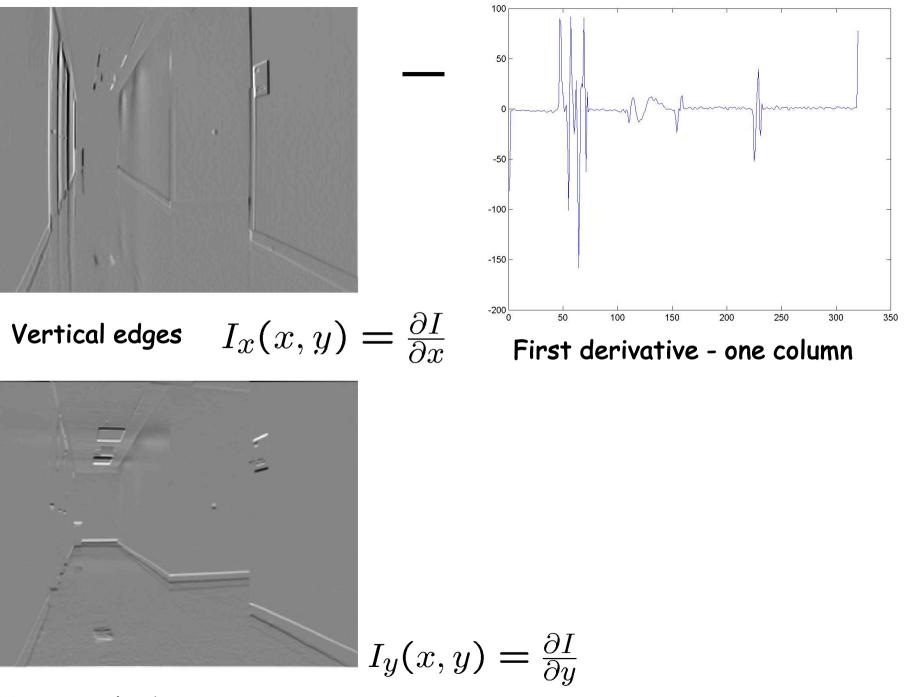
What is the size of the output?



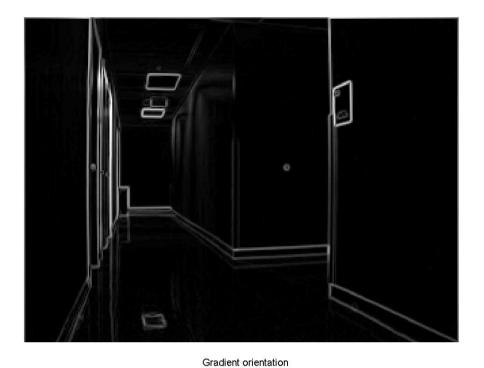
Derivative of Gaussian filter



Which one finds horizontal/vertical edges?



Horizontal edges





• Image Gradient

$$\nabla I = \left[\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}\right]$$

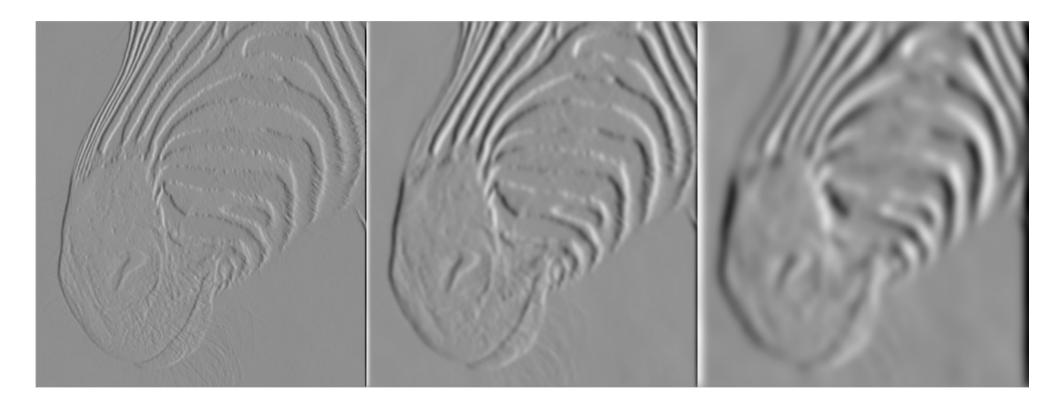
• Gradient Magnitude

$$m = \sqrt{\frac{(\partial I}{\partial x})^2 + (\frac{\partial I}{\partial y})^2}$$

Gradient Orientation

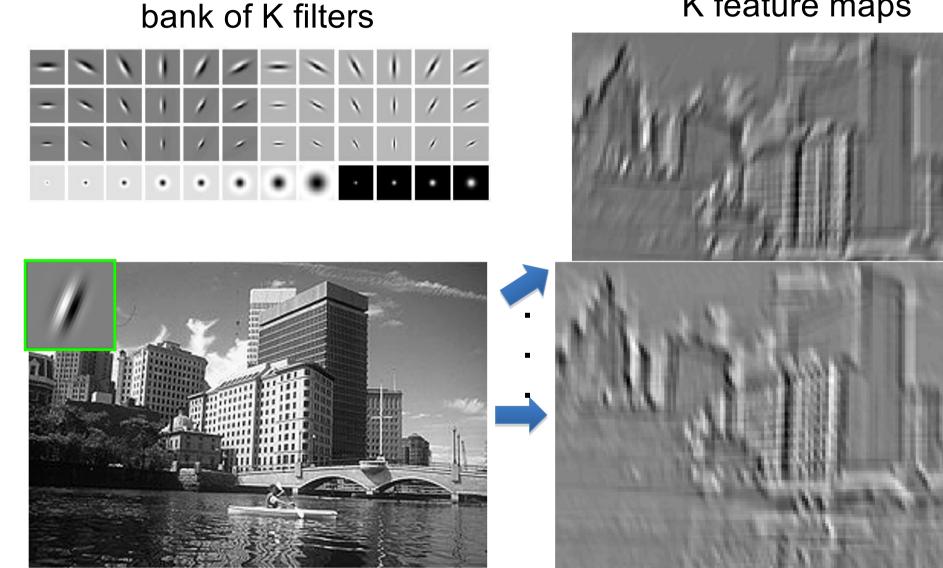
$$\theta = \tan^{-1}(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y})$$

Effect of Smoothing Scale



- Convolution with x-derivative of Gaussian filter with varying scale
- Scale affects the derivative estimates as well as semantics of the edges

Convolution as feature extraction

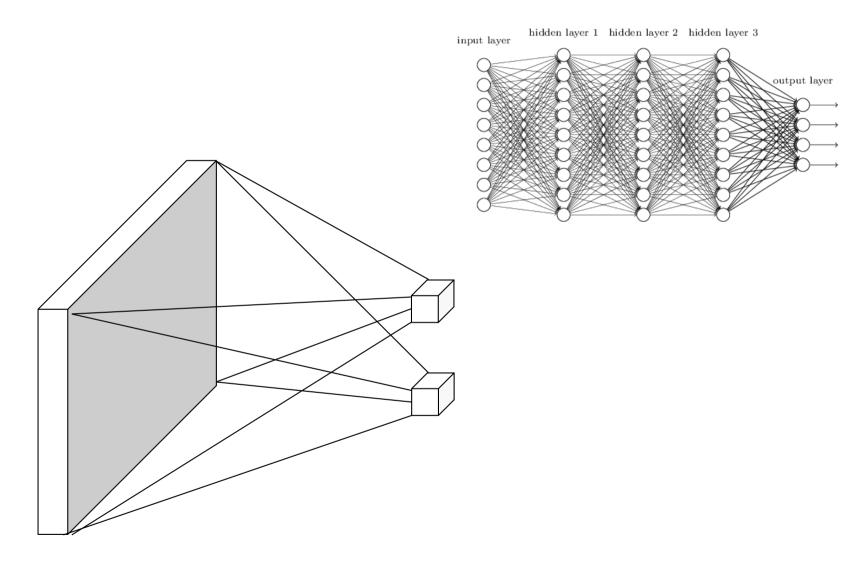


K feature maps

image

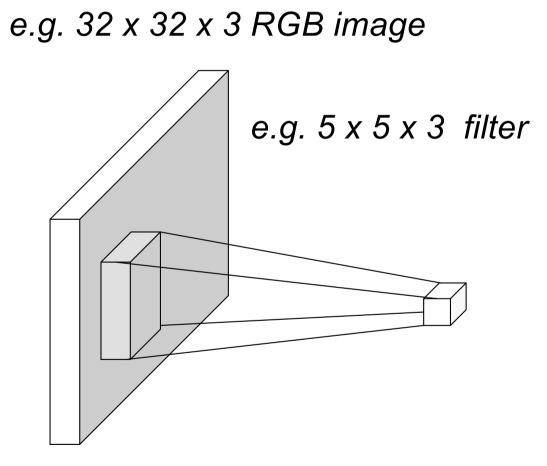
feature map

Neural networks for images



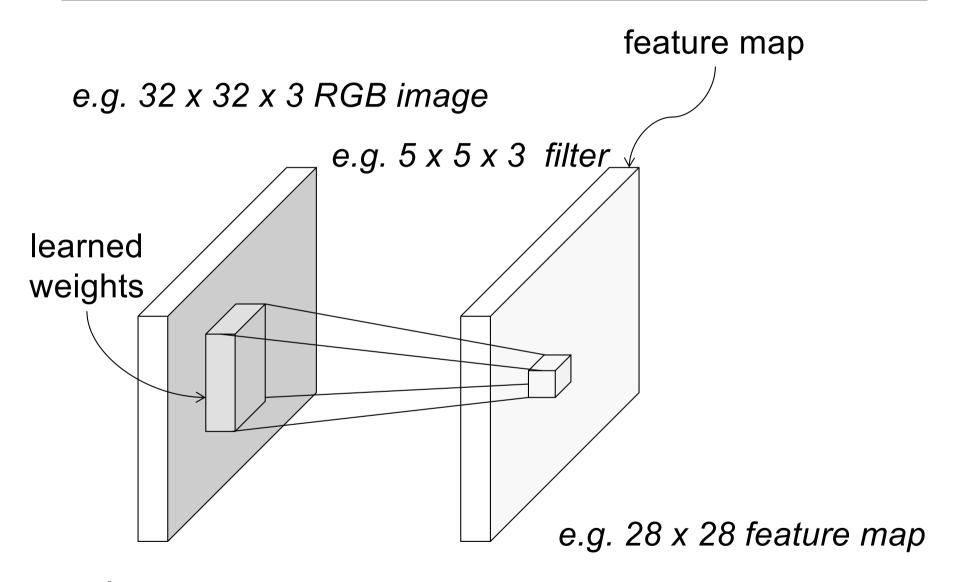
image

Fully connected layer



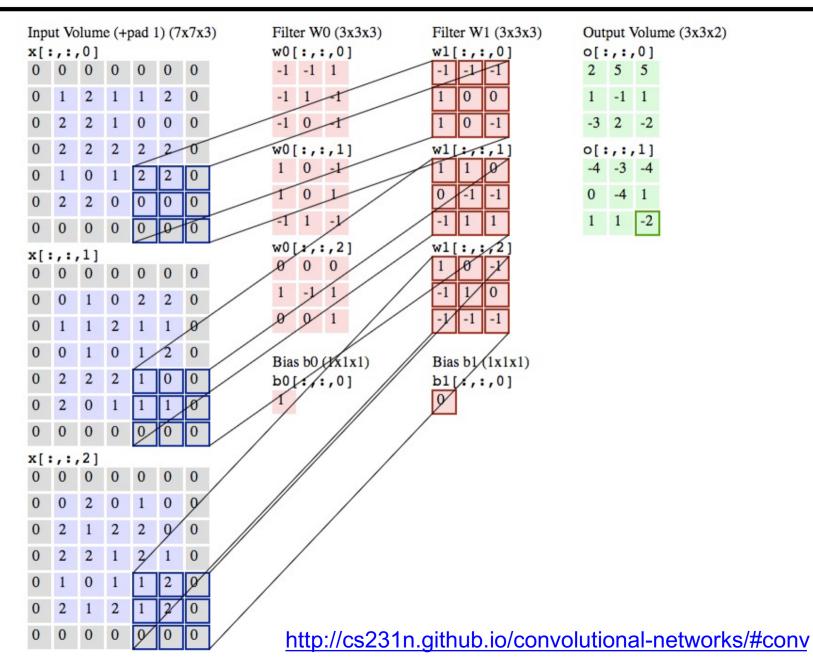
image

Neural networks for images

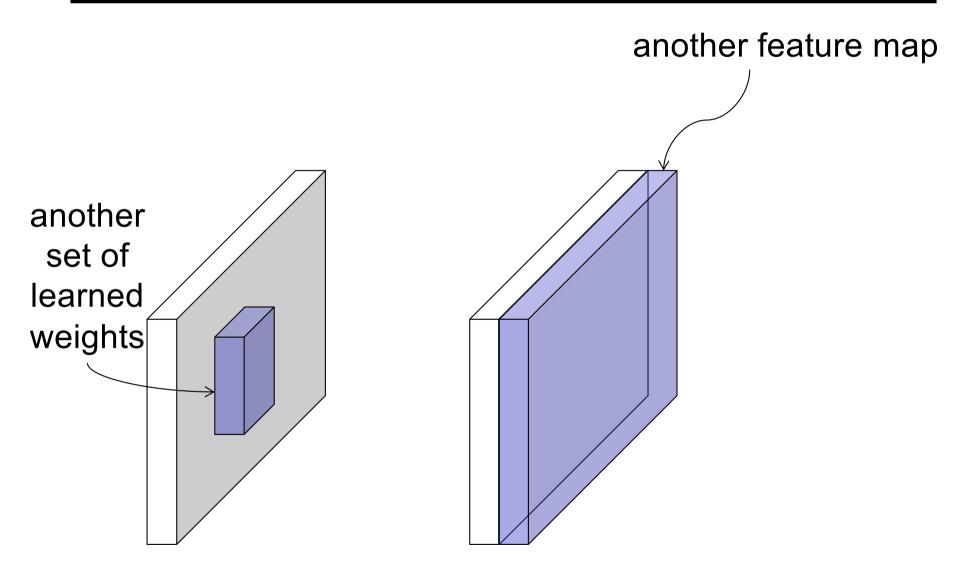


image

Convolutional layer demo



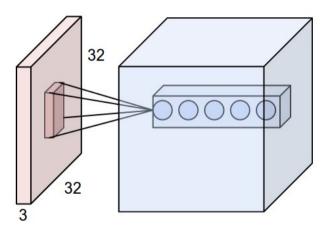
Neural networks for images

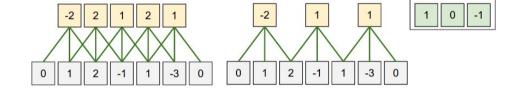


image

Hyperparameters

- size of the filter
- Stride s apply filter at every s-location
- zero-padding

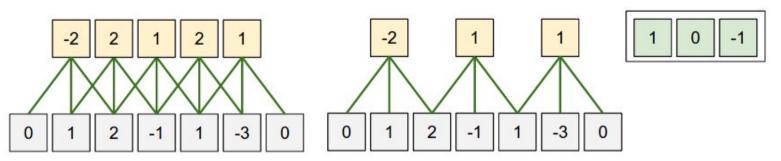




Number of feature maps

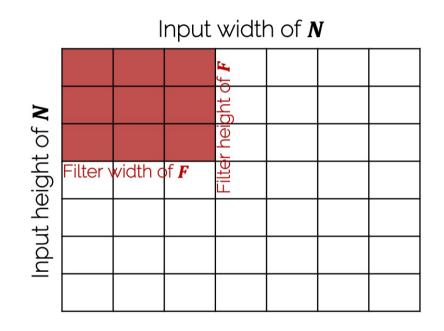
Hyperparameters

- size of the filter
- Stride s apply filter at every s-location
- zero-padding
- E.g. input size 7, filter size 3, stride 1, output size 5
- Input size 7, filter size 3, stied 3, output size 3



- What if the stride is 3 ? Careful does not fit

Stride



Input:	$N \times N$
Filter:	$F \times F$
Stride:	S
Output:	$\left(\frac{N-F}{S}+1\right) \times \left(\frac{N-F}{S}+1\right)$

$$N = 7, F = 3, S = 1: \frac{7-3}{1} + 1 = 5$$

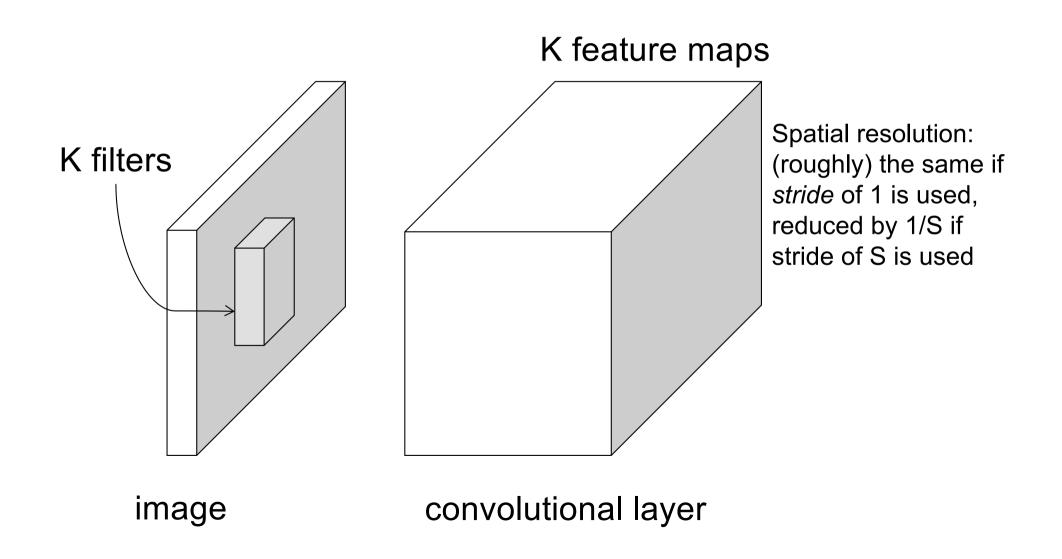
$$N = 7, F = 3, S = 2: \frac{7-3}{2} + 1 = 3$$

$$N = 7, F = 3, S = 3: \frac{7-3}{3} + 1 = 2.\overline{3}$$

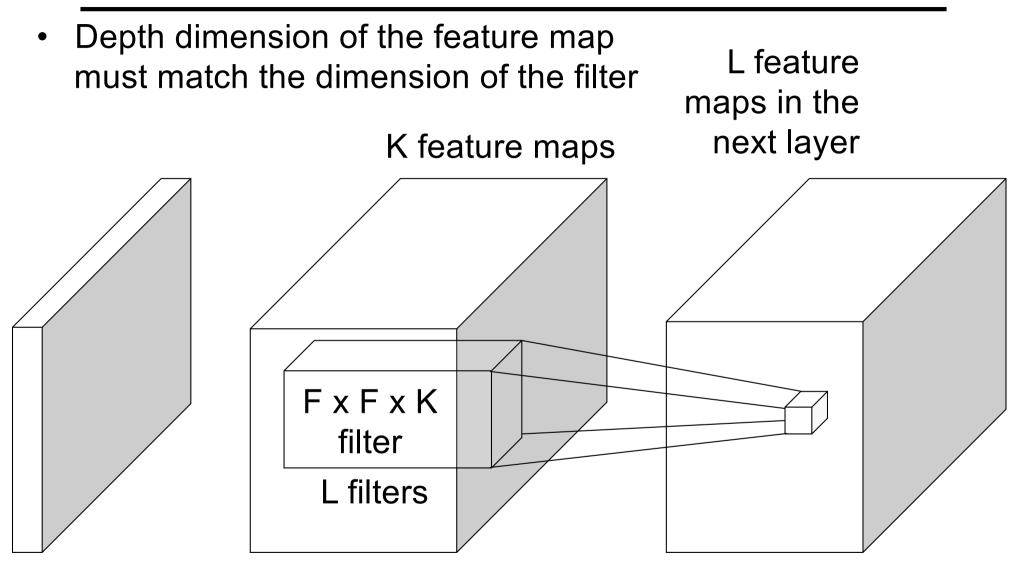
Fractions are illegal

Example from I2DL: Prof. Niessner, Prof. Leal-Taixé

Convolutional layer



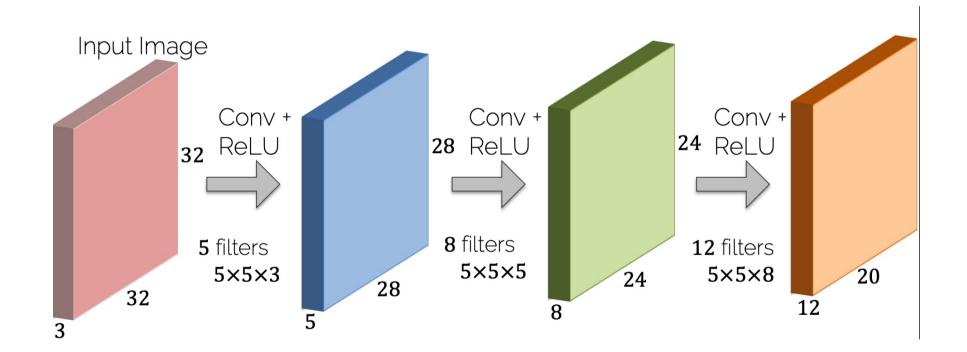
Convolutional layer



image

convolutional layer + ReLU

Stacking up convolutional layers

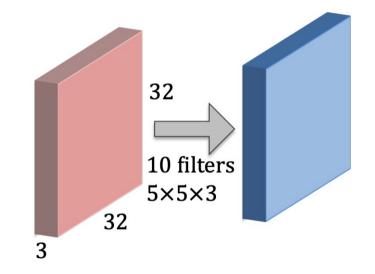


Reduction is size – gets smaller quickly Add padding by 0's

Example from I2DL: Prof. Niessner, Prof. Leal-Taixé

Number of parameters of conv layer





Number of parameters (weights): Each filter has $5 \times 5 \times 3 + 1 = 76$ params (+1 for bias) -> $76 \cdot 10 = 760$ parameters in layer

Example from I2DL: Prof. Niessner, Prof. Leal-Taixé

Pooling layer

Single depth slice of input

3	1	3	5
6	0	7	9
3	2	1	4
0	2	4	3

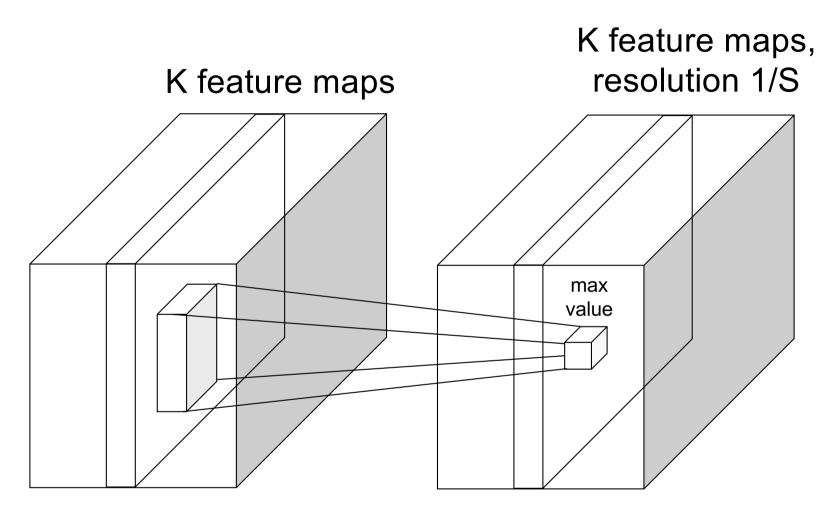
Max pool with **2×2** filters and stride 2



'Pooled' output

6	9
3	4

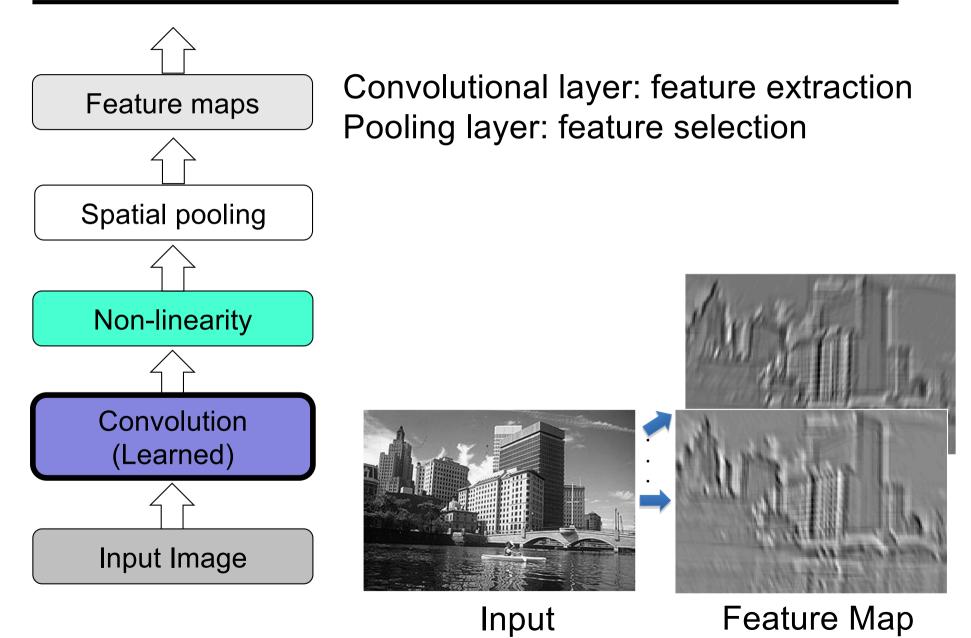
Max pooling layer



F x F pooling filter, stride S Usually: F=2 or 3, S=2

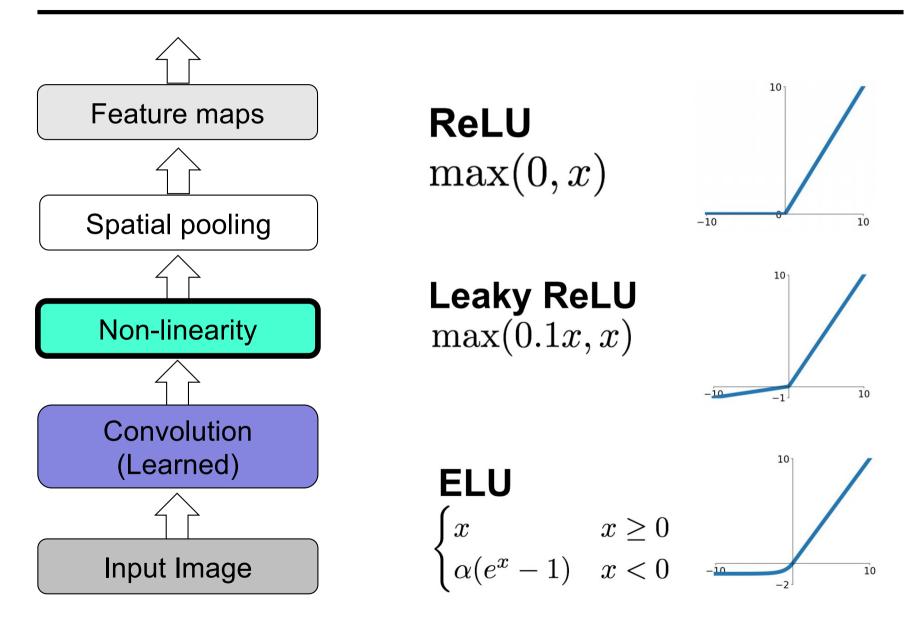
Backward pass: gradient from next layer is passed back only to the unit with max value

Summary: CNN pipeline



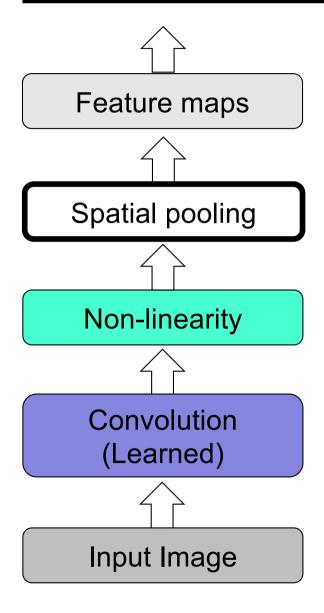
Source: R. Fergus, Y. LeCun

Summary: CNN pipeline

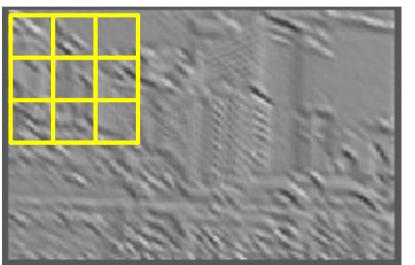


Source: Stanford 231n

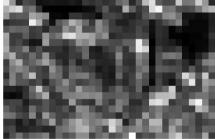
Summary: CNN pipeline



Convolutional layer: feature extraction Pooling layer: feature selection



Max (or Average)



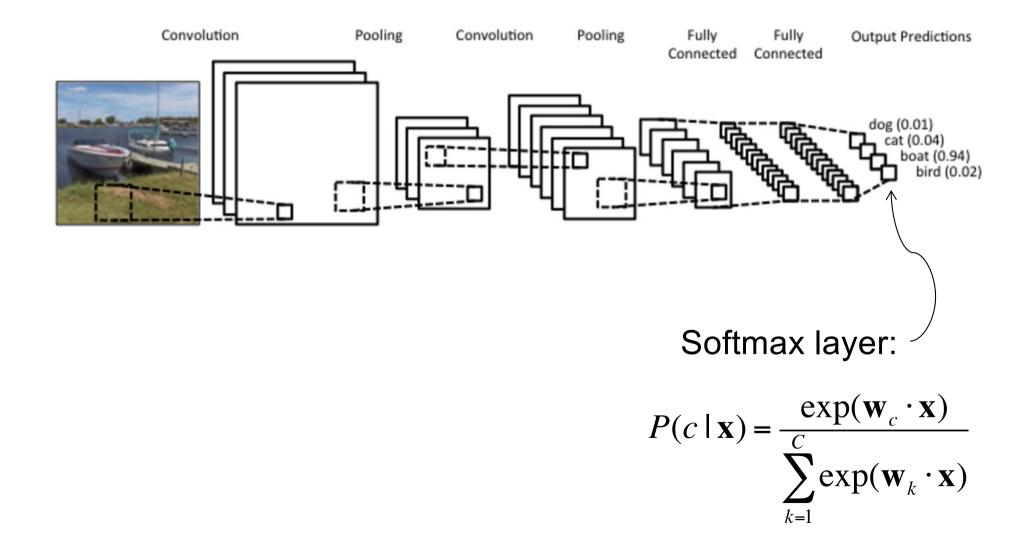
Source: R. Fergus, Y. LeCun

Final Fully Connected FC layer

Connects the feature maps to the final output That makes decisions based on extracted features

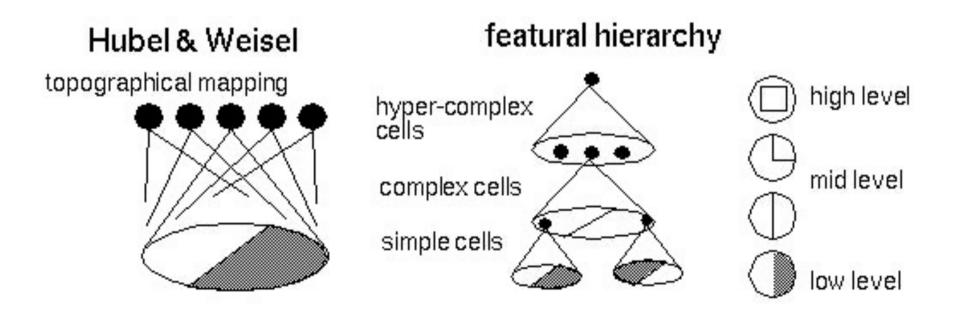
Typically only 1-2 FC layers

Summary: CNN pipeline for classification



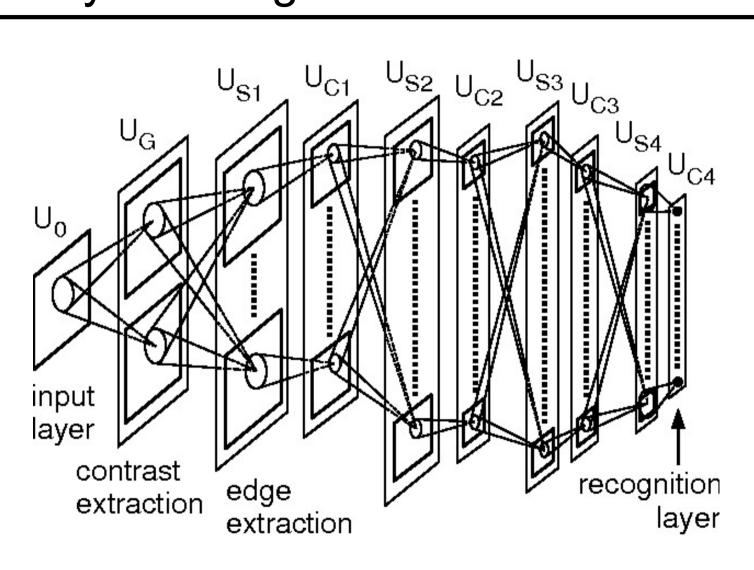
Inspiration: Biological visual system

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



<u>Source</u>

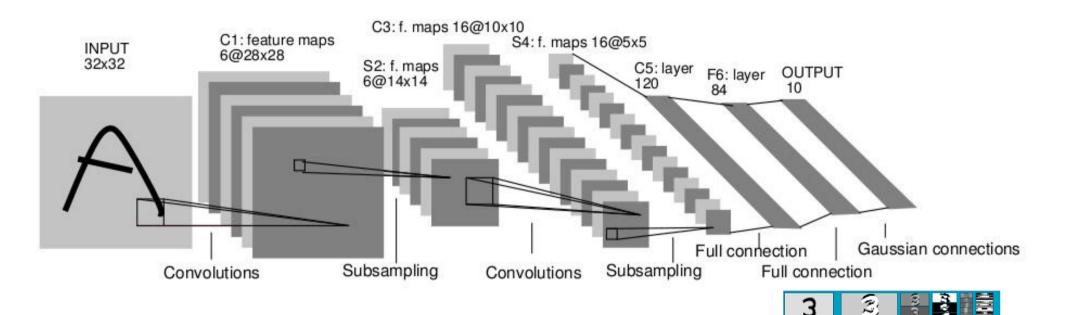
History: Neocognitron



K. Fukushima, 1980s

https://en.wikipedia.org/wiki/Neocognitron

History: LeNet-5



- Average pooling
- Sigmoid or tanh nonlinearity
- Fully connected layers at the end
- Trained on MNIST digit dataset with 60K training examples

3

3

3

8

505

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, <u>Gradient-based learning applied to document</u> recognition, Proc. IEEE 86(11): 2278–2324, 1998.

ImageNet Challenge

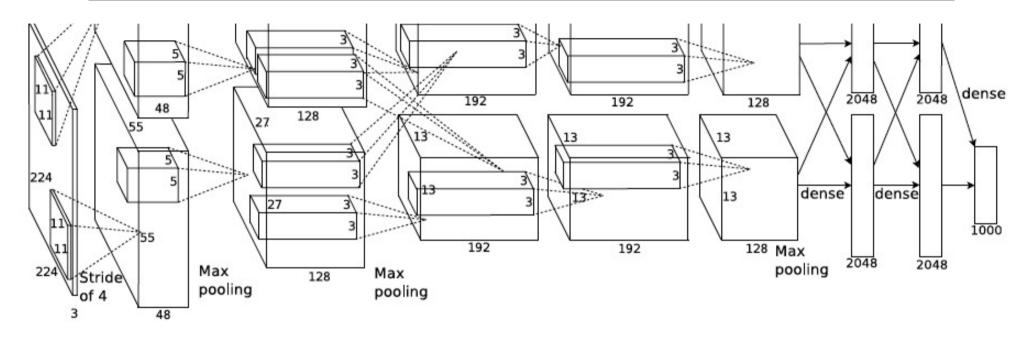
IM GENET



- ~14 million labeled images, 20k classes
- Images gathered from Internet
- Human labels via Amazon MTurk
- ImageNet Large-Scale Visual Recognition Challenge (ILSVRC):
 1.2 million training images, 1000 classes

www.image-net.org/challenges/LSVRC/

AlexNet: ILSVRC 2012 winner



- Similar framework to LeNet but:
 - Max pooling, ReLU nonlinearity
 - More data and bigger model (7 hidden layers, 650K units, 60M params)
 - GPU implementation (50x speedup over CPU)
 - Trained on two GPUs for a week
 - Dropout regularization

A. Krizhevsky, I. Sutskever, and G. Hinton, <u>ImageNet Classification with Deep</u> <u>Convolutional Neural Networks</u>, NIPS 2012

Clarifai: ILSVRC 2013 winner

Refinement of AlexNet

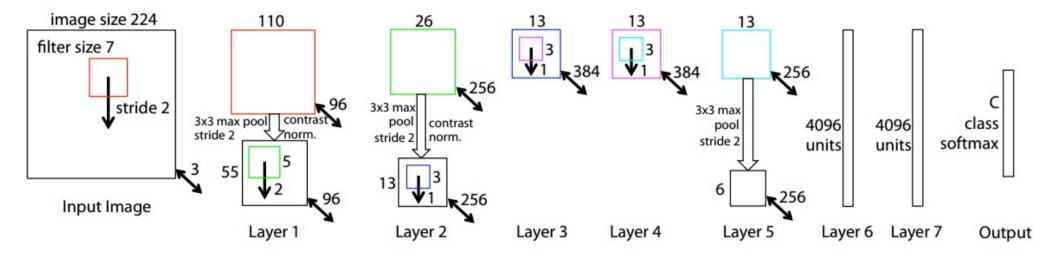
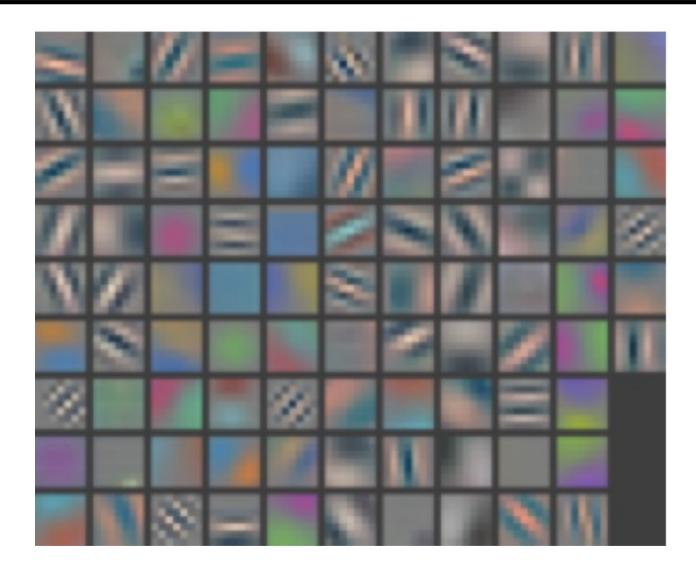


Figure 3. Architecture of our 8 layer convnet model. A 224 by 224 crop of an image (with 3 color planes) is presented as the input. This is convolved with 96 different 1st layer filters (red), each of size 7 by 7, using a stride of 2 in both x and y. The resulting feature maps are then: (i) passed through a rectified linear function (not shown), (ii) pooled (max within 3x3 regions, using stride 2) and (iii) contrast normalized across feature maps to give 96 different 55 by 55 element feature maps. Similar operations are repeated in layers 2,3,4,5. The last two layers are fully connected, taking features from the top convolutional layer as input in vector form ($6 \cdot 6 \cdot 256 = 9216$ dimensions). The final layer is a C-way softmax function, C being the number of classes. All filters and feature maps are square in shape.

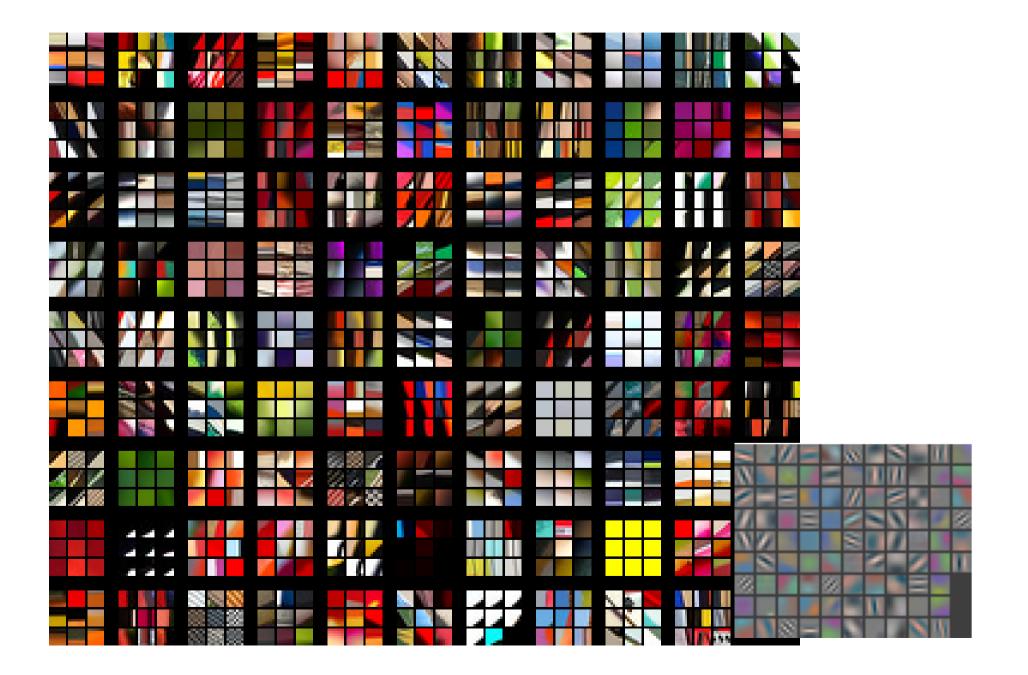
M. Zeiler and R. Fergus, <u>Visualizing and Understanding Convolutional Networks</u>, ECCV 2014 (Best Paper Award winner)

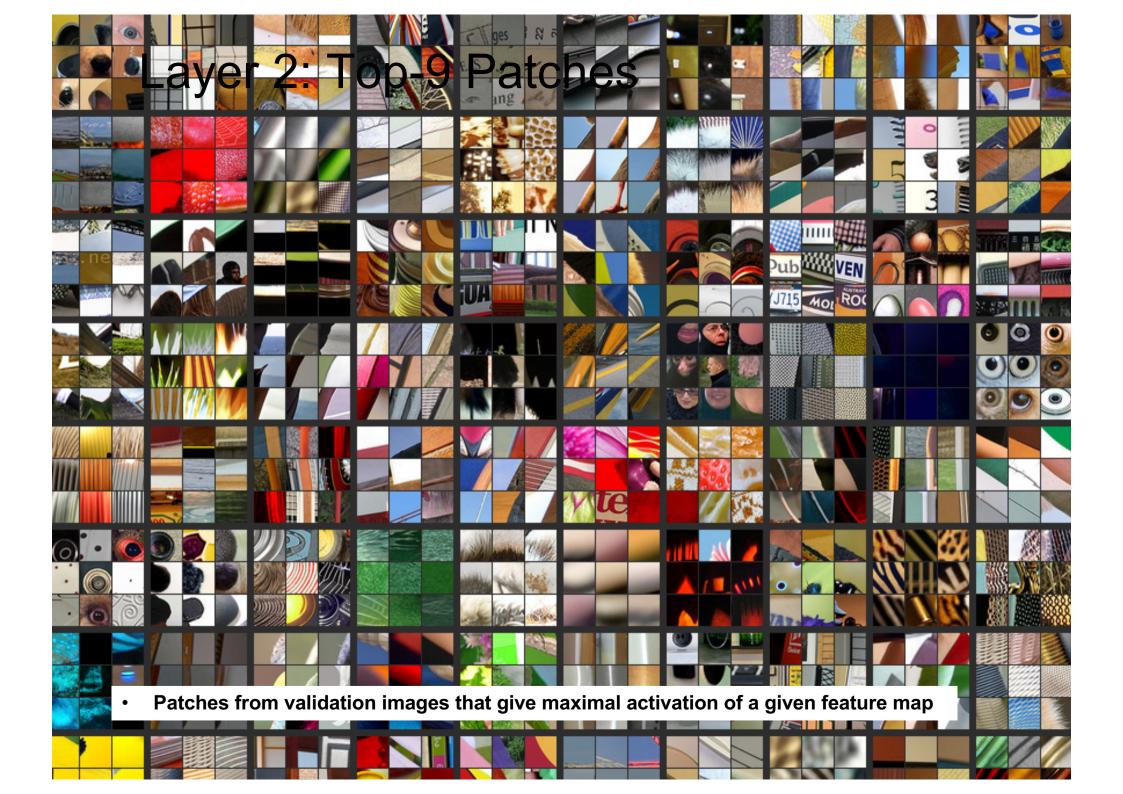
Layer 1 Filters

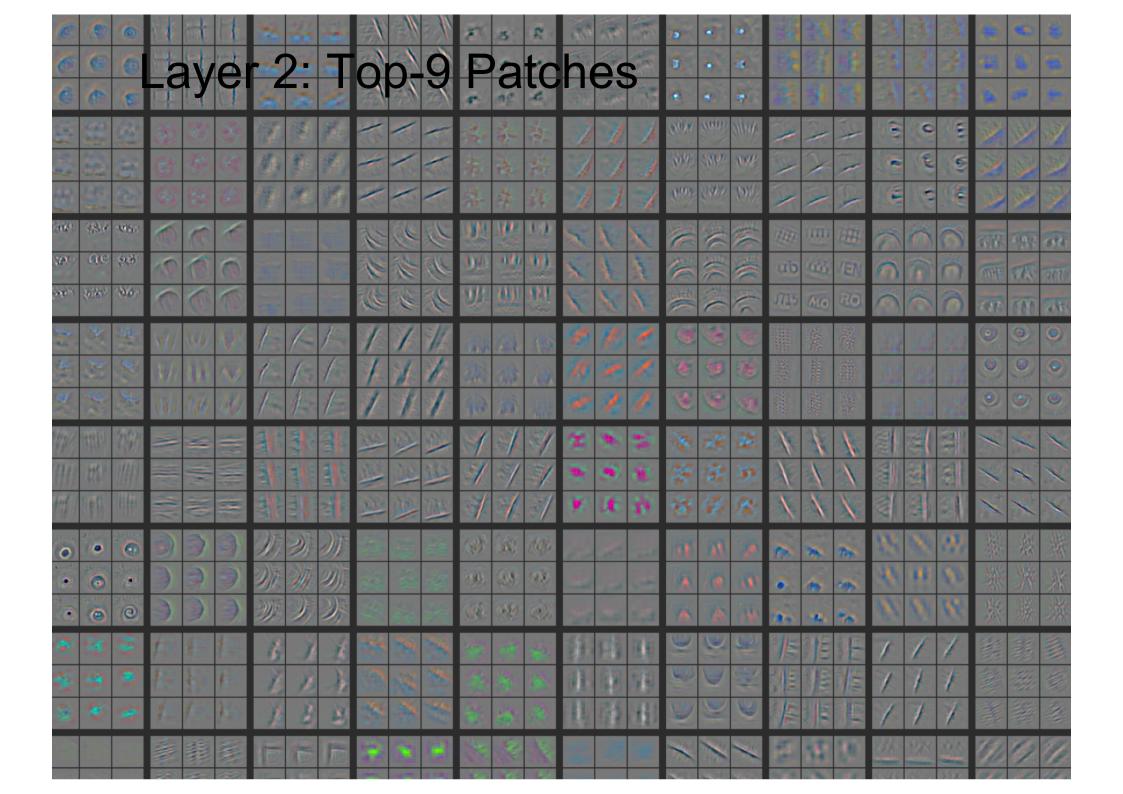


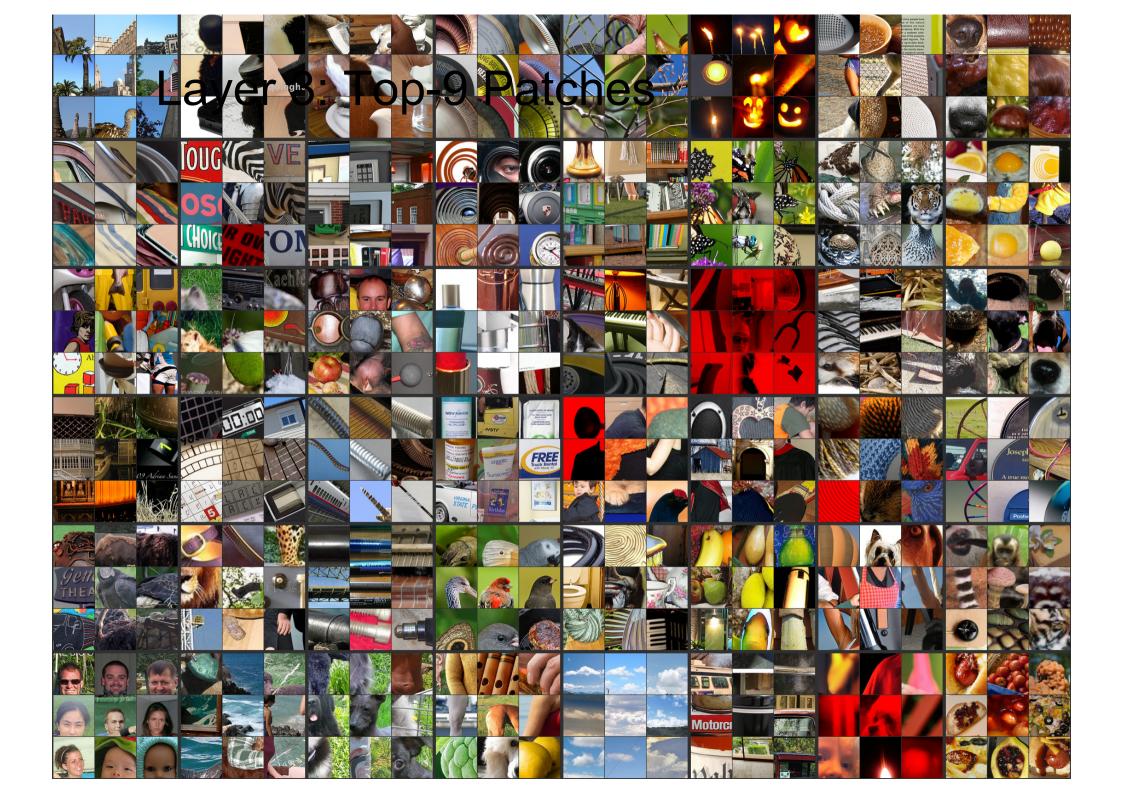
M. Zeiler and R. Fergus, <u>Visualizing and Understanding Convolutional Networks</u>, ECCV 2014 (Best Paper Award winner)

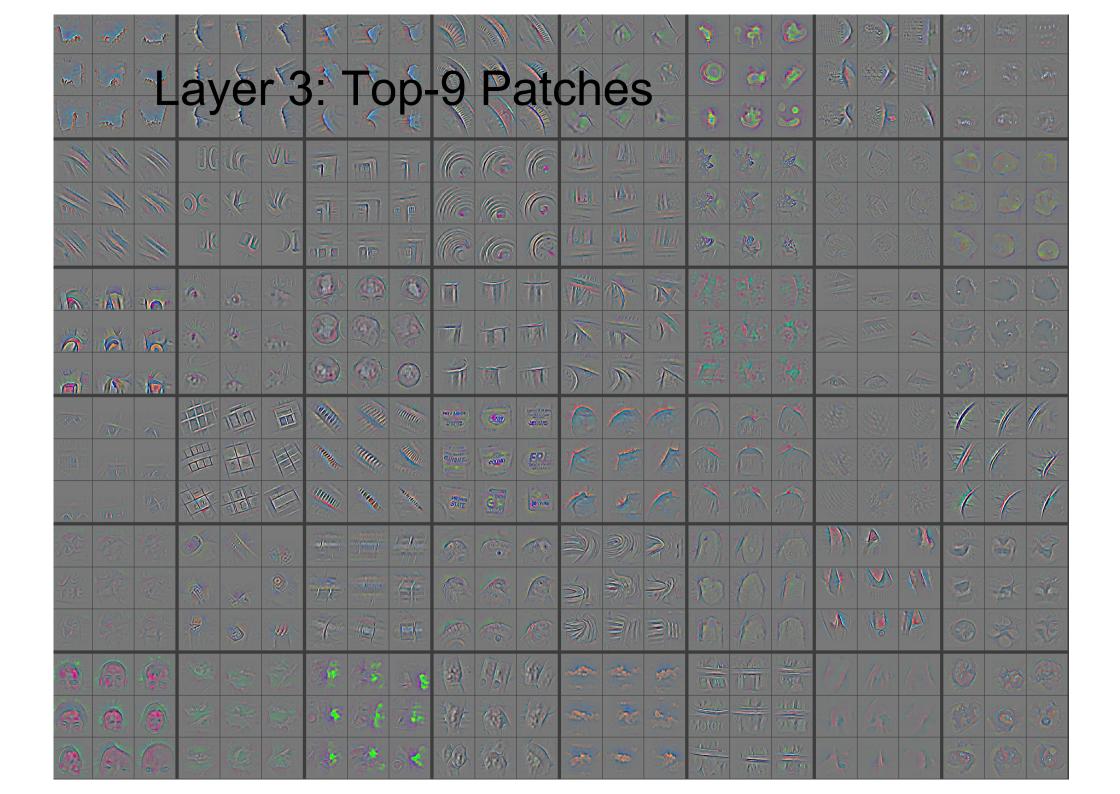
Layer 1: Top-9 Patches

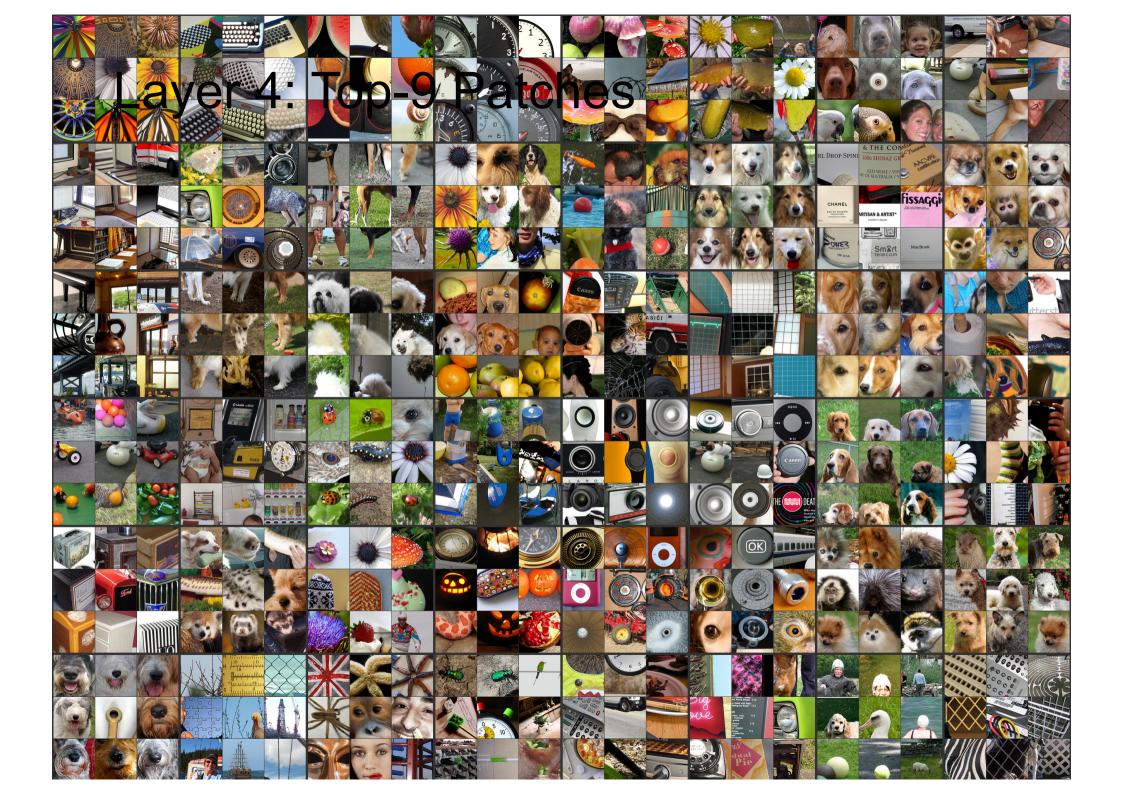




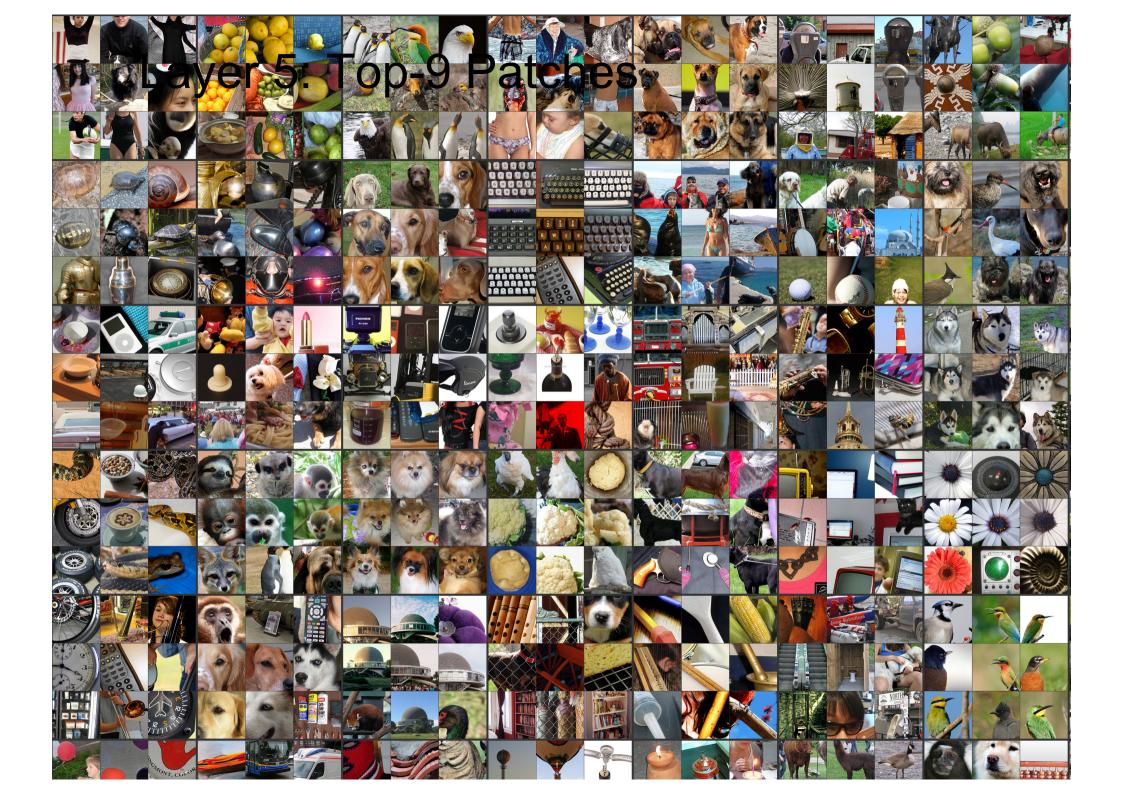








	The second							K.		2	12		Carlo Carlo				Xe	6	63		ANG.	Notes -	AND NO
		a			1	T	n	_0		a	tc	ne	S		Solo and a second					- Contraction of the second se) A		A
	730						r			A CONTRACTOR			Ń		1.33%		-	e	0		X	17	27
	-			Ð.		*		jų.	A pri	and the second sec	-	*		ġł.	6		(33)	DKOP SP	CD WINE	ALO MA	-		000
	No.	ALL CONTRACTOR					X	M	-	- Ali		-			8	20	10	CHANEL LANGE FORTHE	SAN & ART	SAC	000		
與		1				÷.			-	R	- Ale			0	100	6	0	O LESS	Smart	MacBook	3	Q	
			15	J.F.	- 1	2 .				.	0	10	(a)	- And			Ŧ						
R	Q	T	×		J.	Within	Ser Min			000	(The second seco			6			H				3	Ņ	-
1031		P	a construction of the second s	ext.	3.2	an and	and the second s											((Second	X	
-	S.	<u>:</u>]				Ø	00	9		S	N.		0			۲	ALENY DO	<u>(65</u>				Pers	- Maria
٢		05				0		in the second se	X	ý.	М.				0			60	60	<u> </u>	Ŵ	AT .	
3		<u> </u>					1		À.		\$		Ø	0		0				()	APA	1. and	
	Y	W.		Ŷ	<u>J.</u>	())	Ö					0	0		O								1
	A CONTRACTOR	Ĩ	and the second s	20										0		0	Ø						(3)
Y	X						10					0	0	0			0			00	×6	198	
			X	(ngayAfra tanatista	$\langle \langle \rangle$			A Color		¢.	9	0	2. 6. (2.)	-MC	(File			6		Ŕ		000000000000000000000000000000000000000	C.C.
	0		and the second sec		A.	No.	N.		St.	\$		100			and the			6		Ó			100
		(2)	- Julye			XX	N.				<u>.</u>					A	(1) (1) (1)			TT2	MAR		

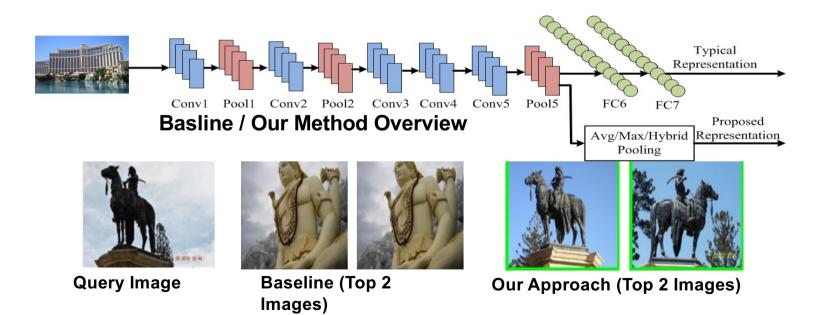


		No.	Ø			1990								E		THE REAL PROPERTY IN THE REAL PROPERTY INTO THE REAL PR				
			ye	ræ			D_6)	at	Ch	es		1		-				(A)	
1 Contraction of the second se								×.			1.00					Ê				E
															(T)				R	
													-		Å	<i>(</i> 3)	(3)		Â	Sec.
								N.	389	000					ð					3
	0	G.		- (1)	Â	*-016F											÷			
					S.			Carlos	8		- Art					, free and the second s			~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	
<u>G</u> e	Ð						BMR		2	95	1				(())					
	 A state 	2				000	20	Ċ,		- Alexandre	(a)					1	Ìr	Ø		Ó
	E			8			1000 m								Î	1	Ē	ø	-	1
				tin		1				A.	app.				ſ	F	TE	Ó		0
31)	- Miles												12		35.	(Å		A		T
(6 <u>6</u>)									*	*	(A)	a start		Net C		1	1	P	- A	
	3	.	Acy								101				(313)					
		M					<i>©</i>						19			Ę		Ŧ		Ř

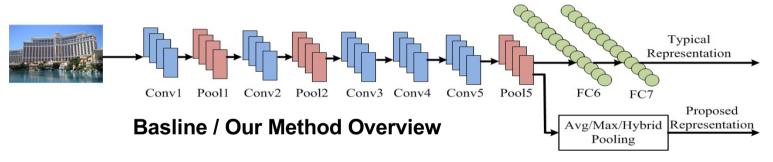
Semantic Matching and Image retrieval

How is the semantic and appearance information represented in the higher convolutional layers on CNN trained for recognition Application: Retrieve images with similar semantics and appearance if the exact instance is not available

A. Mousavian, J. Kosecka Deep Convolutional Features for Image Retrieval and Scene Categorization, 2015



- How is the semantic and appearance information represented in the higher convolutional layers on CNN trained for recognition
- Retrieve images with similar semantics and appearance if the exact instance is not available





Query Image



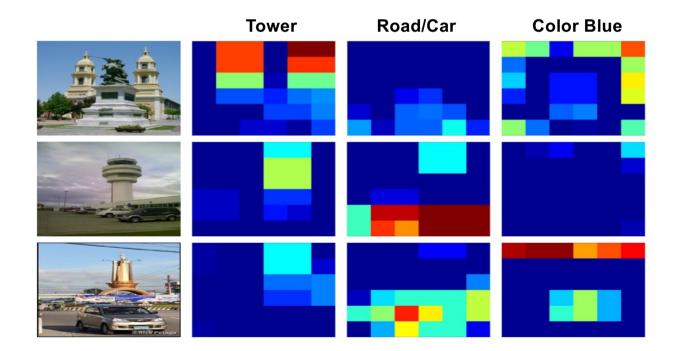
Baseline (Top 2 Images)



Our Approach (Top 2 Images)

Deep Embedding for Image Retrieval

Last convolutional layer has spatial support for semantic categories (different feature activation maps)



Spatial and Magnitude Correlation

The magnitude and location of the objects has high correlation with the scale and location of the object.



Towers



Image representation = 256 dim vector where each element is average of the response map for each channel.



Query Image

Our Method (Average pooling)

Baseline

Example: Max/Hybrid Pooling From Last Convolutional Layer

Image representation = 256 dim vector where each element is the maximum of the response map for each channel.



Quantitative Results on INRIA Holiday Dataset

Method	Dim.	mAP
FC7 (Places CNN)	4096	70.24
FC7 (ImageNet CNN)	4096	68.30
Gong et al. [5] (ImageNet CNN)	12288	80.18
Max pooling (Places CNN)	256	73.72
Max Pooling (ImageNet CNN)	256	70.45
Avg Pooling (Places CNN)	256	76.72
Avg Pooling (ImageNet CNN)	256	73.21
Hybrid Pooling (Places CNN)	512	79.24
Hybrid Pooling (ImageNet CNN)	512	76.34
Max pooling + PCA (Places CNN)	256	77.21
Max Pooling + PCA (ImageNet CNN)	256	76.21
Avg Pooling + PCA (Places CNN)	256	82.86
Avg Pooling + PCA (ImageNet CNN)	256	81.22
Hybrid Pooling + PCA (Places CNN)	512	80.11
Hybrid Pooling + PCA (ImageNet CNN)	512	79.39

Table 1. Evaluation on the INRIA Holiday Dataset with respect to mAP and feature dimensionality

Example: Scene Similarity in the Wild

 Images are semantically labeled using ten semantic classes – sky, building, tree, mountain, road, waterbody, ground, floor, furniture and wall
 Color, texture, geometry and perspective features are extracted from each semantic region and clustered

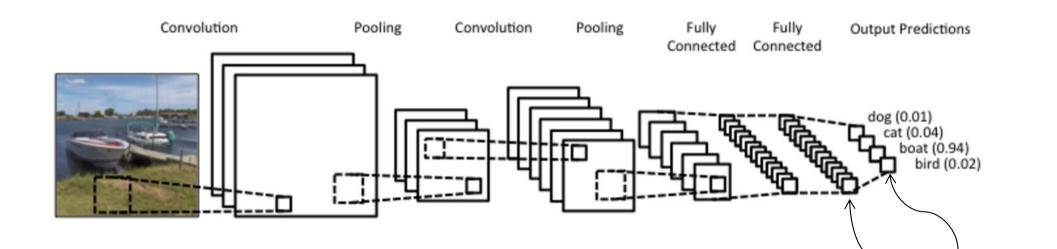


Scene recognition http://places.csail.mit.edu/

What's missing from the picture?

- Training tricks and details: initialization, regularization, normalization
- Training data augmentation
- Averaging classifier outputs over multiple crops/flips
- Ensembles of networks
- Officially, starting with 2015, image classification is not part of ILSVRC challenge, but people continue to benchmark on the data

How to use a trained network for a new task?



Classifier

laver

FC

vector

- Take the vector of activations from one of the fully connected (FC) layers and treat it as an off-the-shelf feature
- Train a new classifier layer on top of the FC layer
- Fine-tune the whole network