Convolutional neural nets: basics

Glenn Bruns
CSUMB

Much material in this deck from Géron, Hands-on Machine Learning with Scikit-Learn and TensorFlow

Learning outcomes

After this lecture you should be able to:

- explain the biological motivation for convolutional neural nets
- define the concepts of:
  - convolutional layer
  - padding
  - stride
  - filter
  - feature map
- write the expression defining the output of a neuron in a convolutional layer

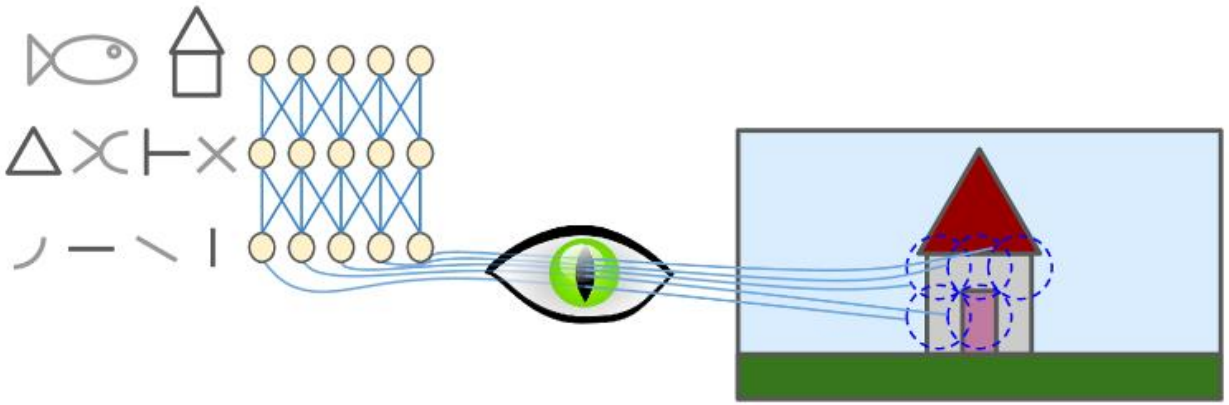
Motivation

- Fully-connected neural nets have issues with image processing:
  - Too many parameters needed for even medium-sized images
  - Recognizing an object in one part of an image doesn't transfer to recognizing in another part
- Similar problems in other domains, like voice recognition and natural language processing

Biological inspiration

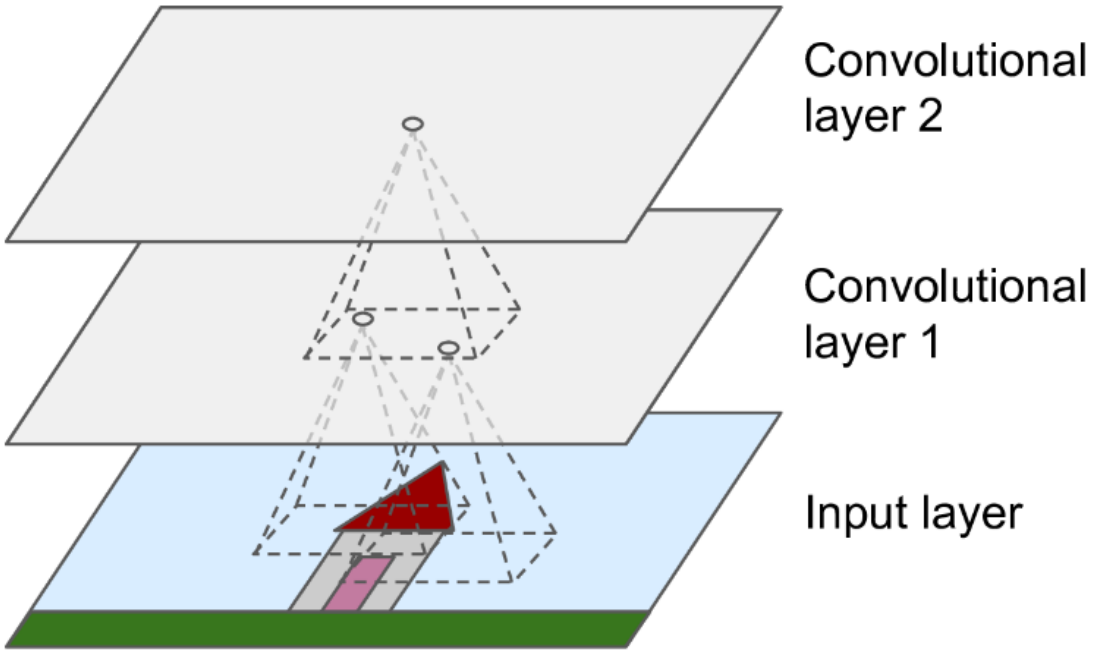
Experiments on cats in 1958/1959 showed:

- many neurons in the visual cortex react only to visual stimuli in a limited region of the visual field
- these "receptive fields" of different neurons overlap
- some neurons react only to horizontal lines, others to vertical lines
- some neurons react to complex combinations of lower-level patterns

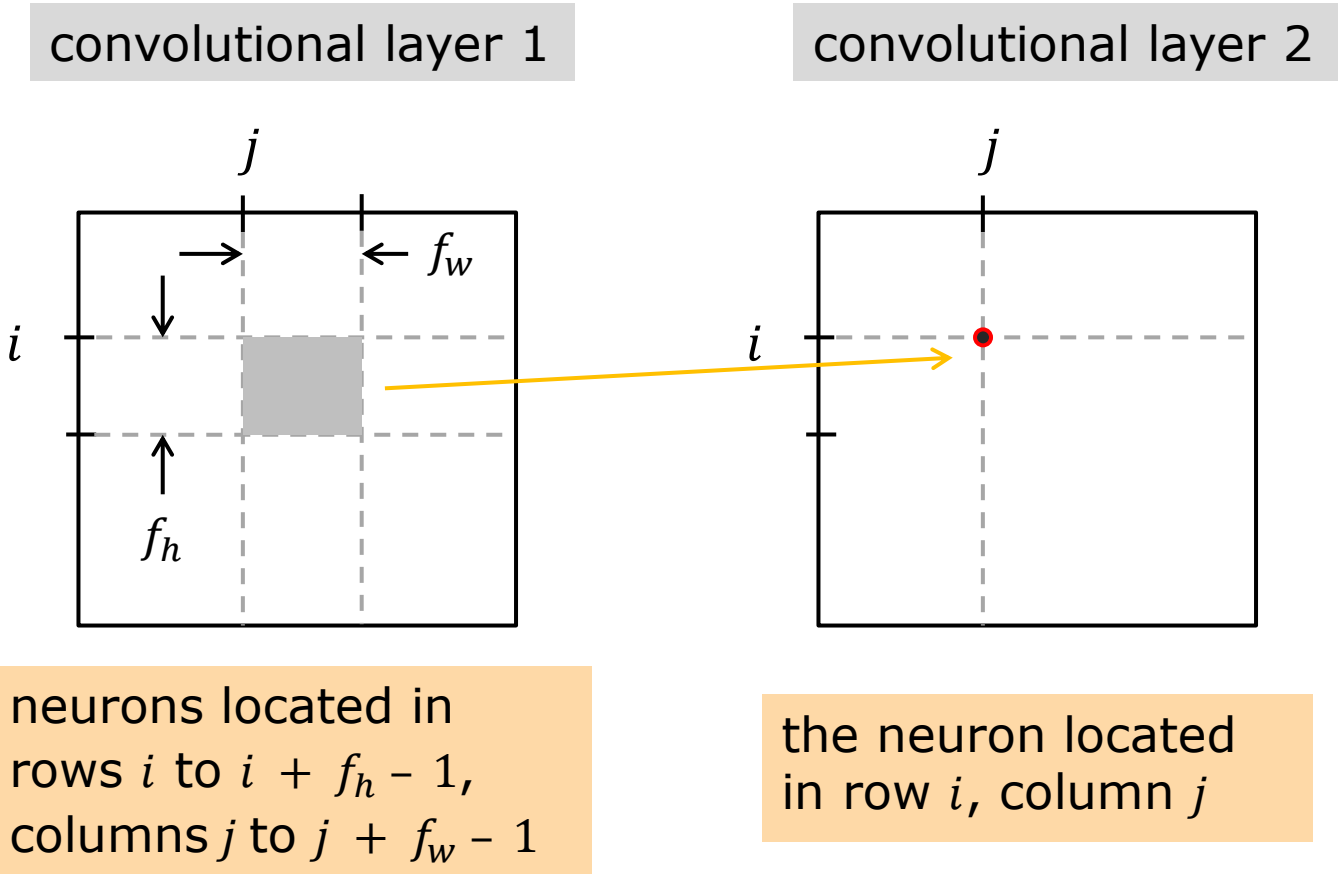


Convolutional layer

- in the first convolutional layer of a Convolutional Neural Network, each neuron "sees" only pixels in its receptive field
- in the second convolutional layer, each neuron is connected only to neurons in a small rectangle of the previous layer



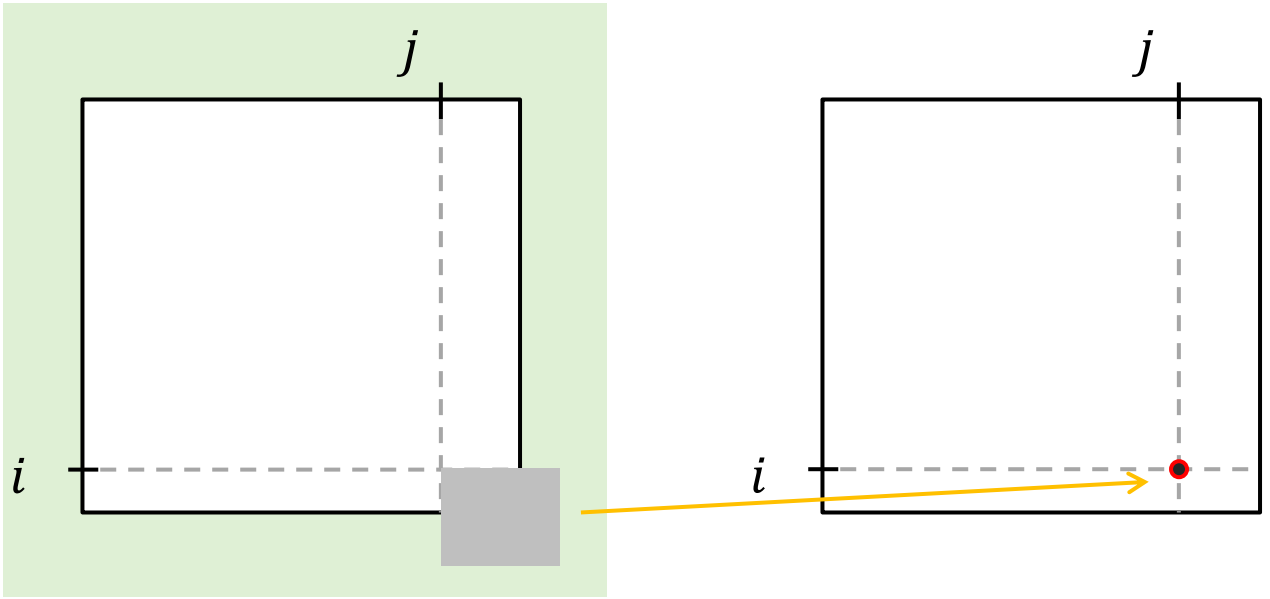
Connecting layers



f_h and f_w are the height and width of the receptive field

Padding

How to deal with this situation?

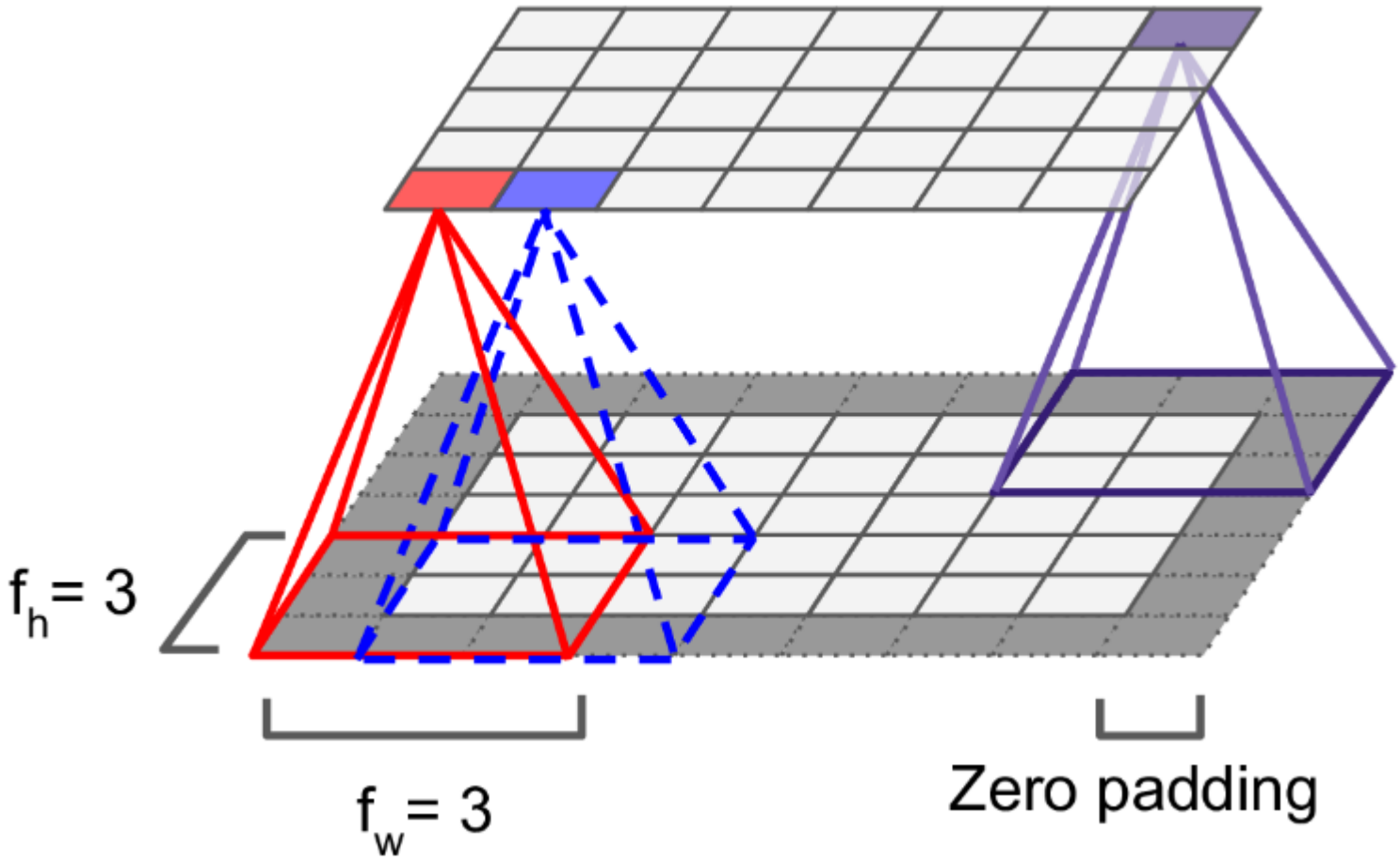


Idea 1: use a second layer that is smaller than the first

Idea 2: add padding around the edge of the first layer

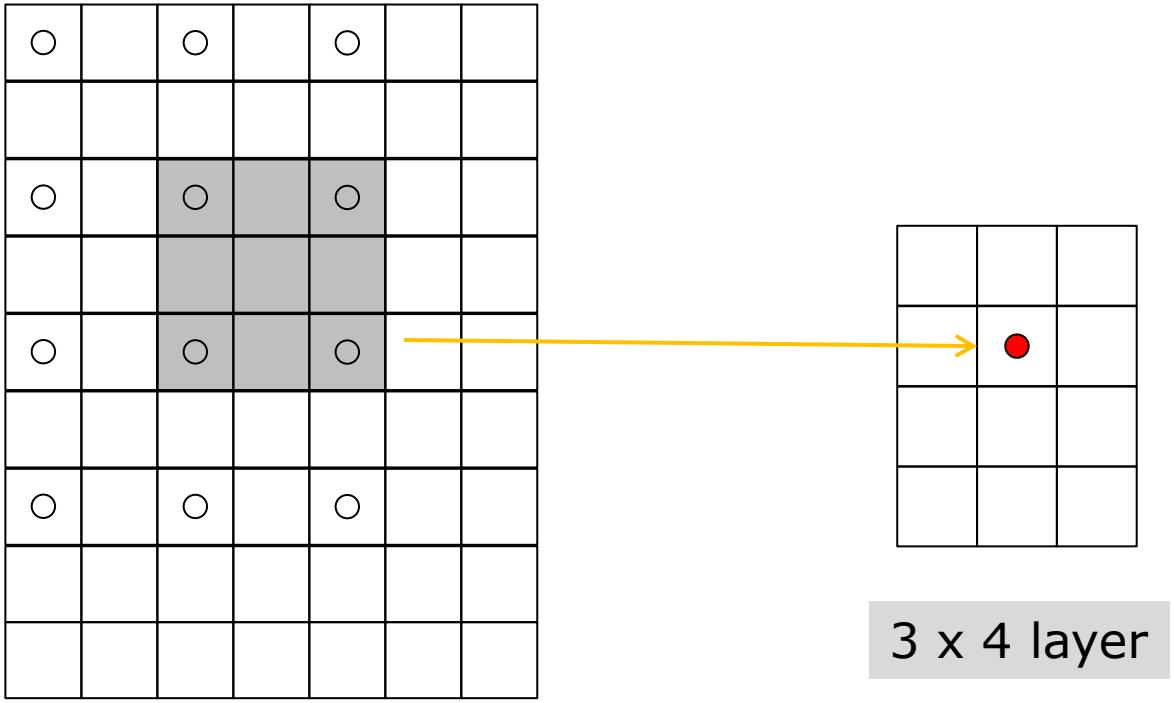
"zero padding"

Padding



Stride

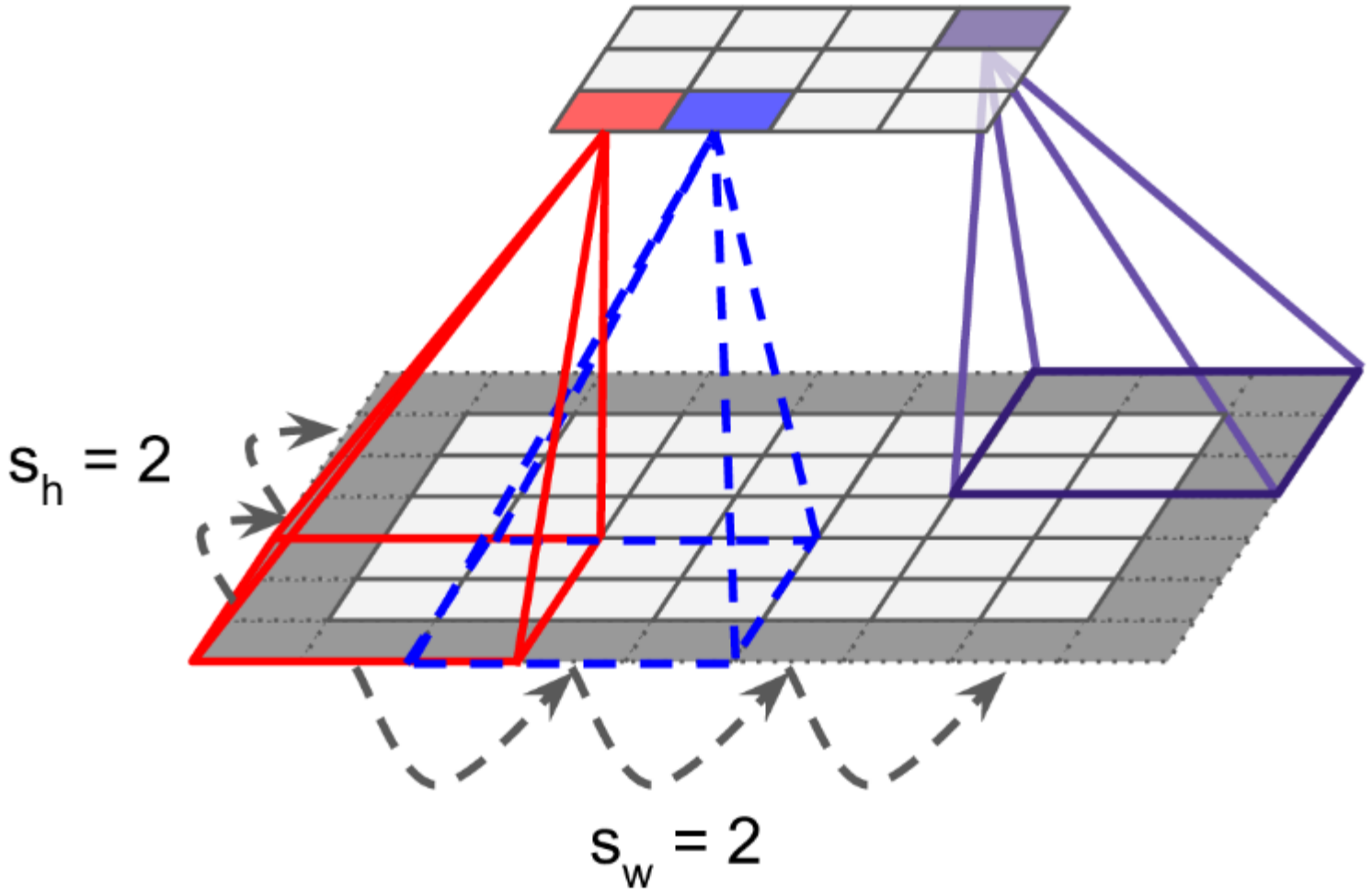
A receptive field can skip over rows and columns of a layer



3 x 4 layer

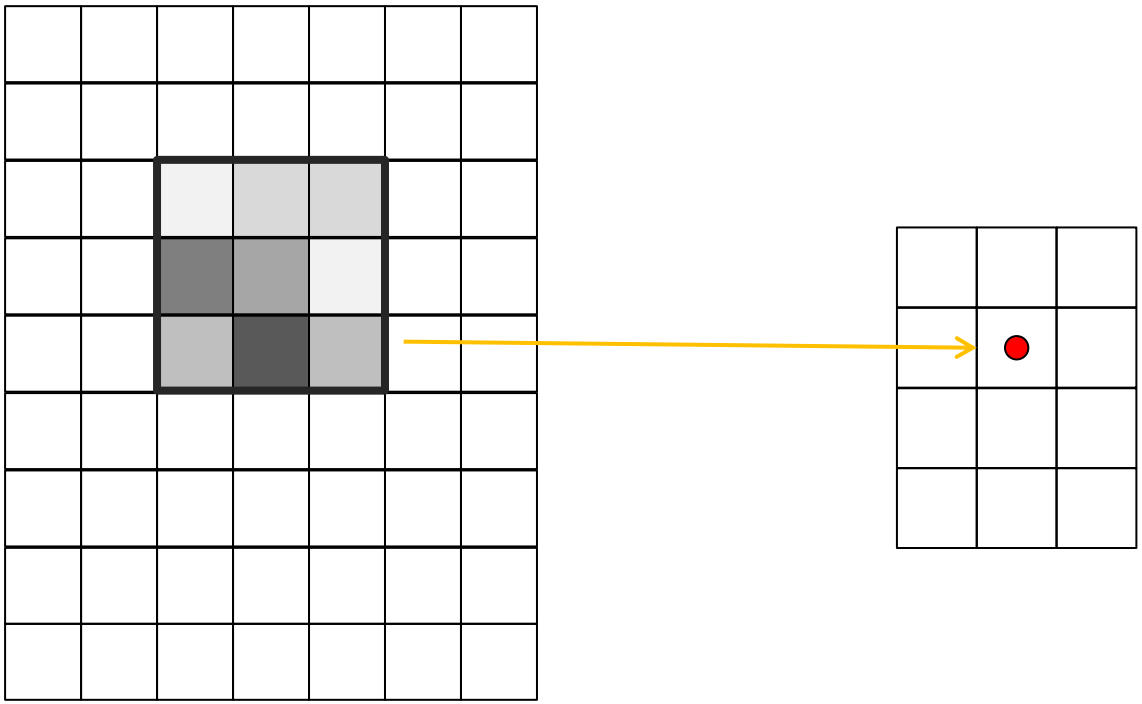
9 x 7 layer
3 x 3 field
stride = 2

Stride



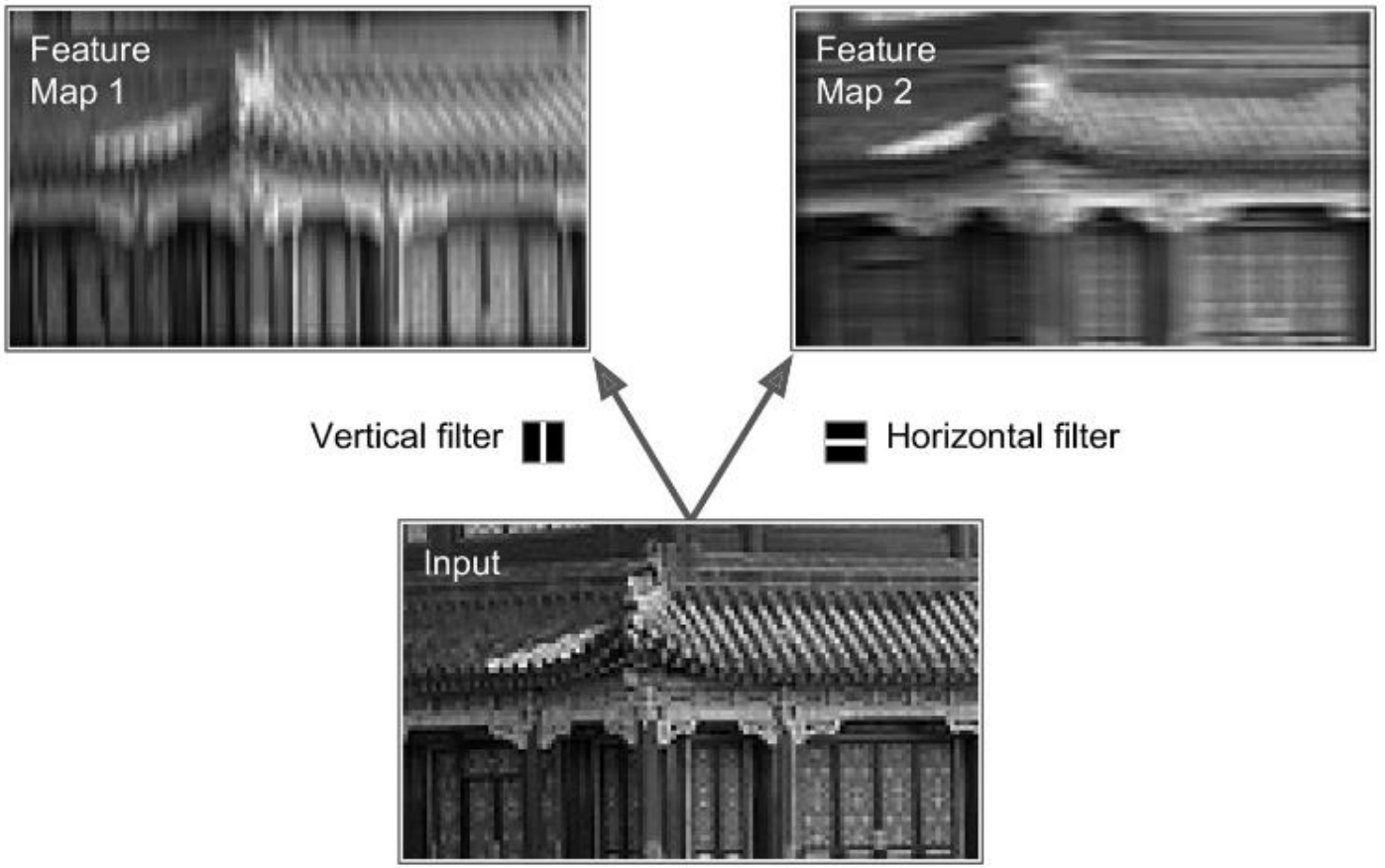
Filters

The weights associated with neuron on the right can be shown as an image – like a heatmap

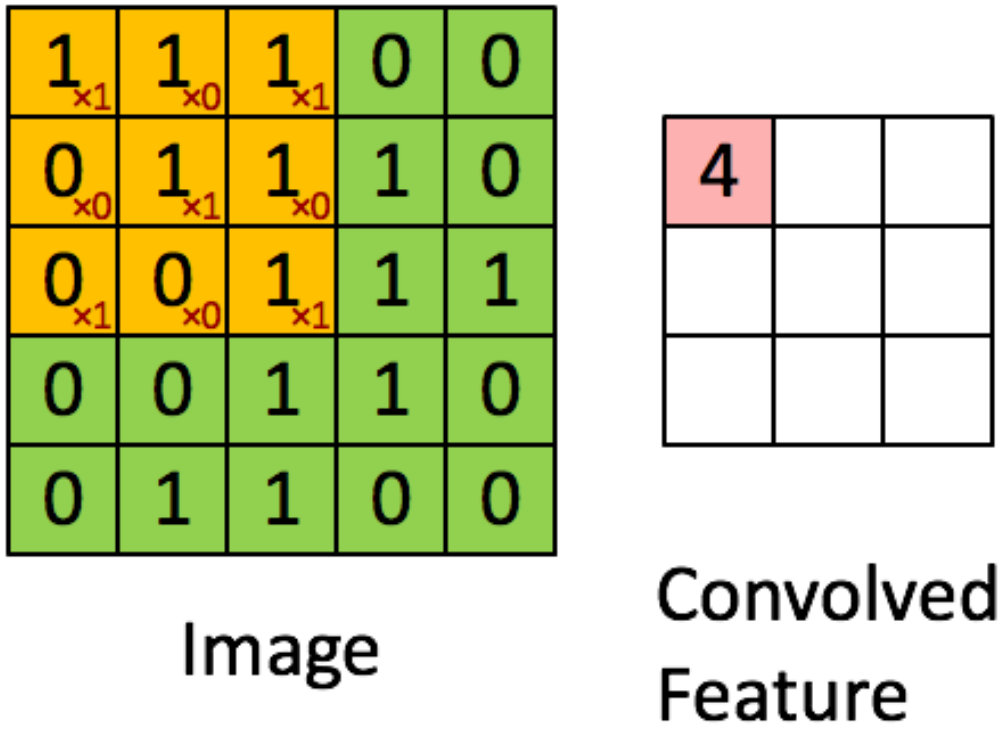


Feature maps

A layer full of neutrons using the same filter is a feature map.



Feature map animation



Orange area is the receptive field for the neuron in pink

source: cv-tricks.com/tensorflow-tutorial/training-convolutional-neural-network-for-image-classification

Feature maps

0	0	0	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

blurring

edge detection

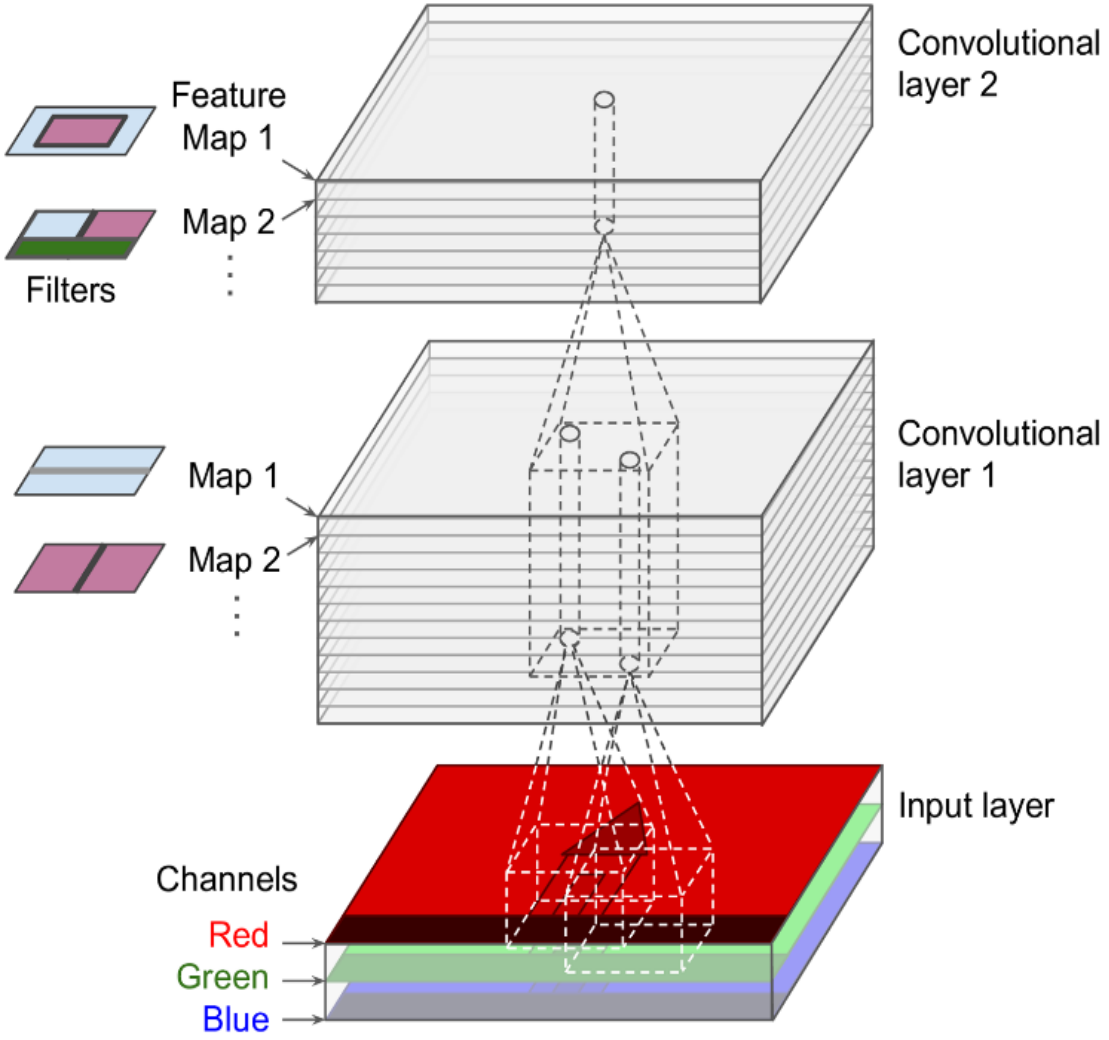
	0	1	0	
	1	-4	1	
	0	1	0	

source:
wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/

Stacking feature maps

- Now the structure of nets is really different.
- We stack feature maps.
  - within each map, all neurons share the same parameters (weights and bias terms)
  - different feature maps can have different parameters
- The receptive field is the same for all the feature maps in a layer.

Stacking feature maps



a convolutional layer applies multiple filters to its inputs

a convolutional layer can then detect multiple features anywhere in its inputs

Computing the output of CNN neuron

$$z_{i,j,k} = b_k + \sum_{u=0}^{f_h-1} \sum_{v=0}^{f_w-1} \sum_{k'=0}^{f_{n'}-1} x_{i',j',k'} . w_{u,v,k',k} \quad \text{with} \begin{cases} i' = i \times s_h + u \\ j' = j \times s_w + v \end{cases}$$

$z_{i,j,k}$ is the output of the neuron at row i, column j in feature map k of the convolutional layer

s_h, s_w are the vertical and horizontal strides

f_h, f_w are the height and width of the receptive field

$f_{n'}$ is the number of feature maps in the previous layer

$x_{i',j',k'}$ is the output of the nuron located in the previous layer at row i', column j', feature map k'

b_k is the bias term for feature map k of this layer

$w_{u,v,k',k}$ is the connection weight between a neuron in feature map k of this layer and its input at row u, column v, (relative to neuron's receptive field) and feature map k'

Summary

Convolutional neural nets are very different from the nets we've seen so far. Now:

- the output of a neuron at one level does not go to every neuron in the next level
- the weights of neurons are not all independent (within a feature map the neurons have the same weights)
- each layer consists of multiple feature maps