Deep Learning & Applications

Vicenç Gómez & Anders Jonsson

Lecture 3: Computer vision

Convolutional neural networks

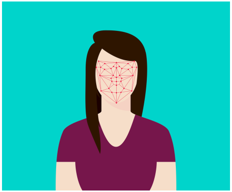
Residual neural networks

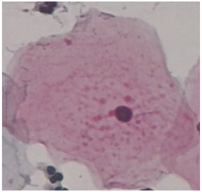
Lecture 3: Computer vision

Convolutional neural networks

Residual neural networks

Image data



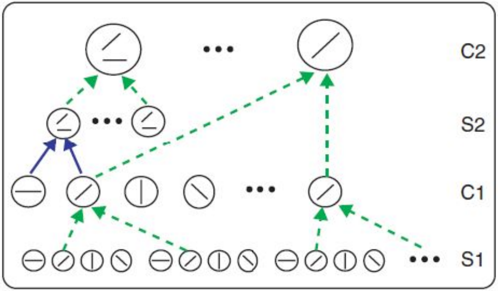


- Many applications are based on visual input
- How do we adapt neural networks to image data?

Input dimensionality

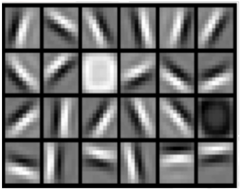
- A common image resolution is $1{,}024 \times 768 = 786{,}432$ pixels
- Each pixel is defined by three colors (RGB)
- 1,000 nodes in first hidden layer $\Rightarrow \geq 2{,}000{,}000{,}000$ weights!

Visual cortex

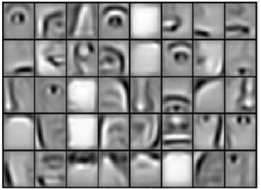


- Cells in visual cortex track features of increasing complexity
- Neurons act as local filters over the input space

From low to high level features



Low-level



Small structure

High-level

Edge detection

$$
\begin{bmatrix} 0 & 0 & 0 & 10 & 10 & 10 \\ 0 & 0 & 0 & 10 & 10 & 10 \\ 0 & 0 & 0 & 10 & 10 & 10 \\ 0 & 0 & 0 & 10 & 10 & 10 \\ 0 & 0 & 0 & 10 & 10 & 10 \\ 0 & 0 & 0 & 10 & 10 & 10 \end{bmatrix}
*
\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}
=
\begin{bmatrix} 0 & -30 & -30 & 0 \\ 0 & -30 & -30 & 0 \\ 0 & -30 & -30 & 0 \\ 0 & -30 & -30 & 0 \end{bmatrix}
$$

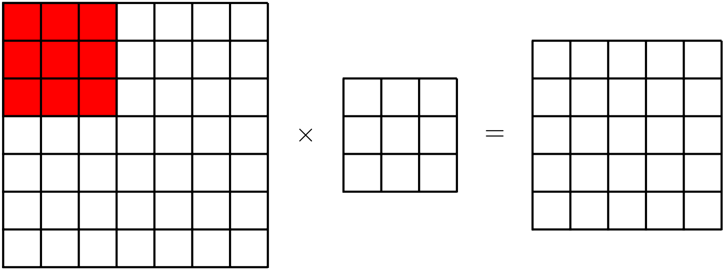
- We can use small filters to detect low-level features
- These filters are repeated over the entire image

Generalized edge detection

$$\begin{array}{|c|c|c|}\hline 1 & 0 & -1 \\ \hline 1 & 0 & -1 \\ \hline 1 & 0 & -1 \\ \hline \end{array} \quad \Rightarrow \quad \begin{array}{|c|c|c|}\hline w_1 & w_2 & w_3 \\ \hline w_4 & w_5 & w_6 \\ \hline w_7 & w_8 & w_9 \\ \hline \end{array}$$

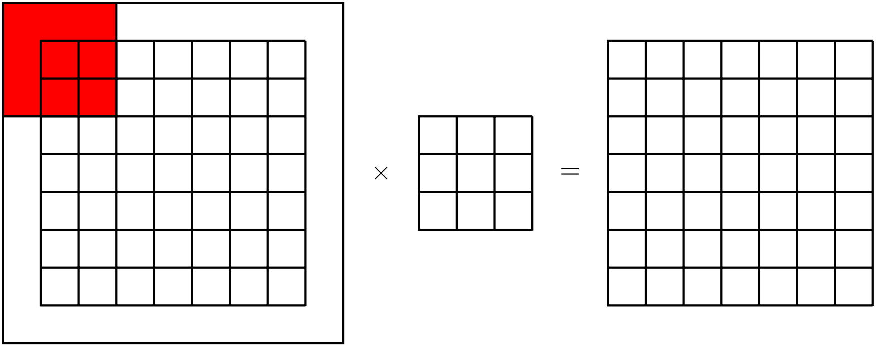
- Use backpropagation to automatically learn filters
- Still repeat the same small filters over the entire image

Shrinking



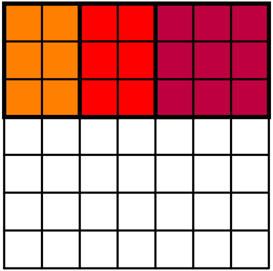
- Applying a filter causes the image size to shrink

Padding



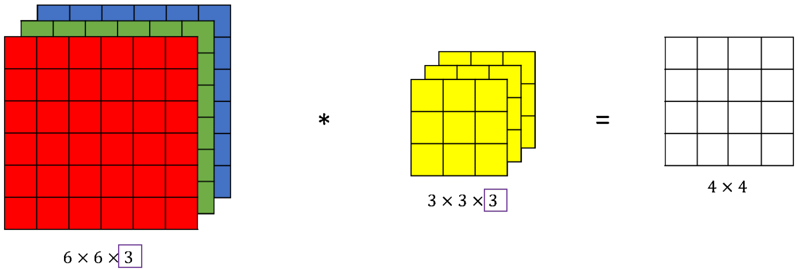
- Padding: add dummy pixels in all directions (usually with value 0)

Stride



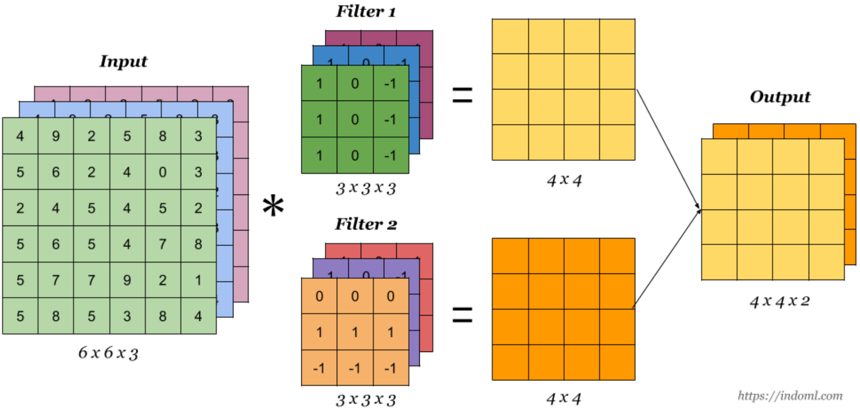
- Stride: how many steps we move the filter in each iteration

Convolution over volumes



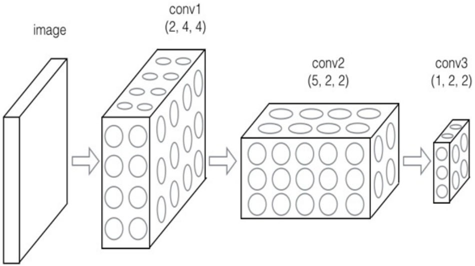
- Filters can be three-dimensional!
- Can capture full color profile of local neighborhoods

Multiple filters



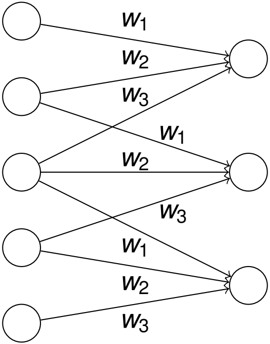
- Typically we apply many filters to the same image

Convolutional neural network (CNN)



- Multiple filters arranged in several layers
- Approximates the mechanism of the visual cortex

Convolutional layer

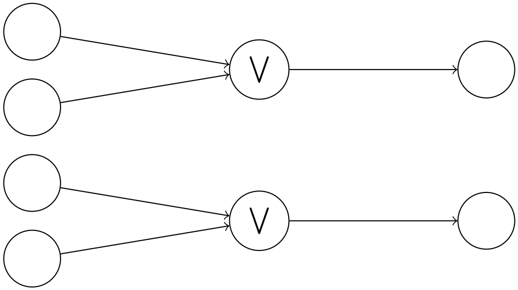


- *Not* fully connected
- Each pattern replicated over the entire input
- Weights are shared!

Input dimensionality

- k filters of size $n \times n \times 3 \Rightarrow 3k \cdot n^2$ weights
- Independent of the number of pixels!
- 100 filters of size $100 \times 100 \times 3 \Rightarrow 3{,}000{,}000 \ll 2{,}000{,}000{,}000$

Subsampling layer

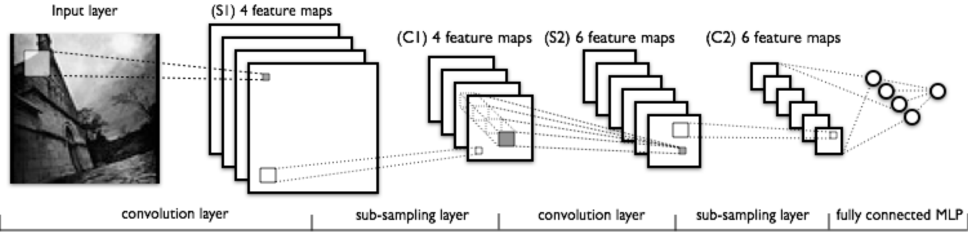


- Reduce number of neurons in layers ($\approx$ increase receptive fields)
- Max-pooling: take the maximum over the inputs

$$o = \max_i x_i$$

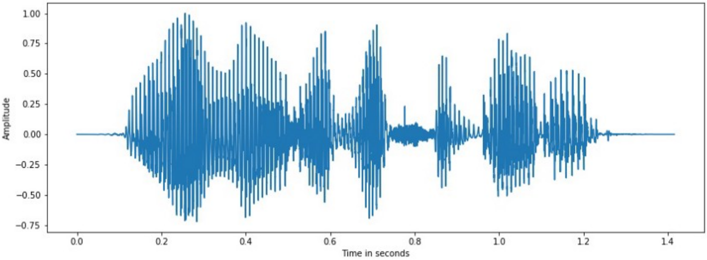
- Error only propagated to the maximum incoming neuron!

Convolutional neural network



- Combine convolutional and subsampling layers
- Often include fully connected layers at the end

Other uses of convolutional networks

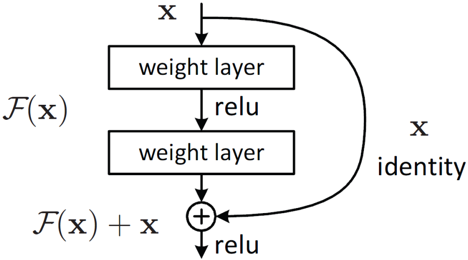


- Somewhat surprisingly, CNNs work well for other inputs
- Applications include speech recognition and natural language analysis

Convolutional neural networks

Residual neural networks

Residual neural network (ResNet)



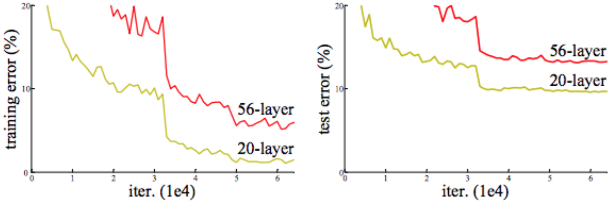
- Feedforward neural network that includes skip connections
- Inspired by pyramidal cells in the cerebral cortex

Barcelona GSE Summer School

Benefits of ResNets

- Alleviates problem of vanishing gradients
- Simplifies the network, initially “skips” some layers
- Explores less of the feature space, learns faster
- Makes it possible to train even deeper networks

Depth of networks



- Usually, performance increases as a function of network depth
- However, there is a point at which performance starts decreasing
- Skip connections avoid performance decrease when adding layers