Learning for vision III
Convolutional networks

Karel Zimmermann
http://cmp.felk.cvut.cz/~zimmerk/

Vision for Robotics and Autonomous Systems
https://cyber.felk.cvut.cz/vras/

Center for Machine Perception
https://cmp.felk.cvut.cz

Department for Cybernetics
Faculty of Electrical Engineering
Czech Technical University in Prague
Outline

- Avoid overfitting by search for the NN model suitable for image processing [Hubel and Wiesel 1960].
- Feedforward and Backprop in ConvNets.
The Tungsten Electrode [Hubel-Science-1957]

http://braintour.harvard.edu/archives/portfolio-items/hubel-and-wiesel

- Device capable to record signal from a single neuron
Experiment with anaesthetised paralysed cat
Recording of electrical signal reveals:
1. Nearby neurons process information from nearby visual fields (topographic map).
2. Neurons with similar function organized into columns.
3. Neurons are sensitive to edges and its orientation.

[Hubel and Wiesel 1959]
[Hubel and Wiesel 1960]

[Hubel and Wiesel 1960]

paralysed cat  
awake monkey

1. Nearby neurons process information from nearby visual fields (topographical map).

- Processing of visual information in cortex is not fully connected.
1. Nearby neurons process information from nearby visual fields (topographical map).

![Diagram showing processing of N-pixel image and n-dimensional spatial neighbourhood.]

- What is dimensionality reduction for N-pixel image and n-dimensional spatial spatial neighbourhood?
2. Neurons with similar function organized into columns

- There are neurons which detect an edge on the left and there are different neurons which detect the same edge on the right.

2. Neurons with similar function organized into columns

\[ \sigma(w_1^\top x_1) \]

\[ \sigma(w_2^\top x_2) \]

\[ \sigma(w_3^\top x_3) \]

\[ \sigma(w_4^\top x_4) \]
2. Neurons with similar function organized into columns

- What is dimensionality reduction for N pixel image and n-dimensional spatial neighbourhood?
2. Neurons with similar function organized into columns

\[
\sigma(w^T \bar{x}_1) \\
\sigma(w^T \bar{x}_2) \\
\sigma(w^T \bar{x}_3) \\
\sigma(w^T \bar{x}_4)
\]

- \(N^2 \text{ vs } n\) (it does not depend on the image resolution)
- It corresponds to convolution of image \(x\) with kernel \(w\) followed by activation function
2. Neurons with similar function organized into columns

- \( N^2 \) vs \( n \) (it does not depend on the image resolution)
- It corresponds to convolution of image \( x \) with kernel \( w \) followed by activation function

\[
\text{conv}(x, w)
\]
3. Neurons are sensitive to edges and its orientation

Inputs which maximized output of \textit{layer 1}

[Zeiler and Fergus, ECCV, 2014]
3. Neurons are sensitive to edges and its orientation.

Inputs which maximized output of layer 2

[Zeiler and Fergus, ECCV, 2014]
Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Cybernetics
3. Neurons are sensitive to edges and its orientation

Inputs which maximized output of layer 3

[Zeiler and Fergus, ECCV, 2014]

Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Cybernetics
3. Neurons are sensitive to edges and its orientation

Inputs which maximized output of **layer 4**

[Zeiler and Fergus, ECCV, 2014]
3. Neurons are sensitive to edges and its orientation.

Inputs which maximized output of **layer 5**

[Zeiler and Fergus, ECCV, 2014]
Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Cybernetics
Hubel and Wiesel experiments in 1950s and 1960s

• Nobel Prize in Physiology and Medicine in 1981
• Dr. Hubel: “There has been a myth that the brain cannot understand itself. It is compared to a man trying to lift himself by his own bootstraps. We feel that is nonsense. The brain can be studied just as the kidney can.”


Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Cybernetics
Convolution forward pass \( y = \text{conv}(x, w) \)
Convolution forward pass \( y = \text{conv}(x, w) \)

\[
\begin{array}{cccc}
  y_{11} & y_{12} \\
  y_{21} & y_{22}
\end{array}
\]

\[
\text{conv} \left( \begin{array}{ccc}
  x_{11} & x_{12} & x_{13} \\
  x_{21} & x_{22} & x_{23} \\
  x_{31} & x_{32} & x_{33}
\end{array} \right), \quad \begin{array}{cc}
  w_{11} & w_{12} \\
  w_{21} & w_{22}
\end{array}
\right)
\]

\[
\begin{align*}
y_{11} &= w_{11}x_{11} + w_{12}x_{12} + w_{21}x_{21} + w_{22}x_{22} \\
y_{12} &= w_{11}x_{12} + w_{12}x_{13} + w_{21}x_{22} + w_{22}x_{23} \\
y_{21} &= w_{11}x_{21} + w_{12}x_{22} + w_{21}x_{31} + w_{22}x_{32} \\
y_{22} &= w_{11}x_{22} + w_{12}x_{23} + w_{21}x_{32} + w_{22}x_{33}
\end{align*}
\]
Convolution forward pass \( y = \text{conv}(x, w) \)

\[
\begin{array}{c|c|c}
    y_{11} & y_{12} & \text{conv} \left( \begin{array}{ccc}
        x_{11} & x_{12} & x_{13} \\
        x_{21} & x_{22} & x_{23} \\
        x_{31} & x_{32} & x_{33} \\
    \end{array} \right),
\end{array}
\]

- \( y_{11} = w_{11}x_{11} + w_{12}x_{12} + w_{21}x_{21} + w_{22}x_{22} \)
- \( y_{12} = w_{11}x_{12} + w_{12}x_{13} + w_{21}x_{22} + w_{22}x_{23} \)
- \( y_{21} = w_{11}x_{21} + w_{12}x_{22} + w_{21}x_{31} + w_{22}x_{32} \)
- \( y_{22} = w_{11}x_{22} + w_{12}x_{23} + w_{21}x_{32} + w_{22}x_{33} \)
Convolution forward pass \( y = \text{conv}(x, w) \)

\[
\begin{array}{c|c|c}
  | x_{11} & x_{12} & x_{13} \\
  \hline
  x_{21} & x_{22} & x_{23} \\
  \hline
  x_{31} & x_{32} & x_{33} \\
\end{array}
\]

\[
\begin{array}{c|c}
  | w_{11} & w_{12} \\
  \hline
  w_{21} & w_{22} \\
\end{array}
\]

\[
\begin{align*}
y_{11} &= w_{11}x_{11} + w_{12}x_{12} + w_{21}x_{21} + w_{22}x_{22} \\
y_{12} &= w_{11}x_{12} + w_{12}x_{13} + w_{21}x_{22} + w_{22}x_{23} \\
y_{21} &= w_{11}x_{21} + w_{12}x_{22} + w_{21}x_{31} + w_{22}x_{32} \\
y_{22} &= w_{11}x_{22} + w_{12}x_{23} + w_{21}x_{32} + w_{22}x_{33} \\
\end{align*}
\]
Convolution forward pass  \( y = \text{conv}(x, w) \)

\[
\begin{array}{cc}
 y_{11} & y_{12} \\
 y_{21} & y_{22} \\
\end{array}
\]

\[
\begin{array}{ccc}
 x_{11} & x_{12} & x_{13} \\
 x_{21} & x_{22} & x_{23} \\
 x_{31} & x_{32} & x_{33} \\
\end{array}
\]

\[
\begin{array}{cc}
 w_{11} & w_{12} \\
 w_{21} & w_{22} \\
\end{array}
\]

\[
y_{11} = w_{11} x_{11} + w_{12} x_{12} + w_{21} x_{21} + w_{22} x_{22} \\
y_{12} = w_{11} x_{12} + w_{12} x_{13} + w_{21} x_{22} + w_{22} x_{23} \\
y_{21} = w_{11} x_{21} + w_{12} x_{22} + w_{21} x_{31} + w_{22} x_{32} \\
y_{22} = w_{11} x_{22} + w_{12} x_{23} + w_{21} x_{32} + w_{22} x_{33}
\]
Convolution forward pass \( y = \text{conv}(x, w) \)

\[
\begin{array}{c|c}
\hline
y_{11} & y_{12} \\
\hline
y_{21} & y_{22} \\
\hline
\end{array}
\quad \text{conv} \quad \begin{bmatrix}
x_{11} & x_{12} & x_{13} \\
x_{21} & x_{22} & x_{23} \\
x_{31} & x_{32} & x_{33} \\
\end{bmatrix}, \quad \begin{bmatrix}
w_{11} & w_{12} \\
w_{21} & w_{22} \\
\end{bmatrix}
\]

\[
y_{11} = w_{11}x_{11} + w_{12}x_{12} + w_{21}x_{21} + w_{22}x_{22} \\
y_{12} = w_{11}x_{12} + w_{12}x_{13} + w_{21}x_{22} + w_{22}x_{23} \\
y_{21} = w_{11}x_{21} + w_{12}x_{22} + w_{21}x_{31} + w_{22}x_{32} \\
y_{22} = w_{11}x_{22} + w_{12}x_{23} + w_{21}x_{32} + w_{22}x_{33}
\]
Convolution layer properties - output size

\[
\text{conv} \left( \begin{array}{c}
\text{image} \\
(5\times5)
\end{array} \right), \quad \begin{array}{c}
\text{kernel} \\
(2\times2)
\end{array} \right) = \begin{array}{c}
\text{output} \\
(\_ \times \_)
\end{array}
\]
Convolution layer properties - output size

\[
\text{conv} (\text{image} (5\times5), \text{kernel} (2\times2)) = \text{output} (? \times ?)
\]
Convolution layer properties - output size

\[
\text{conv} \left( \begin{array}{c}
\text{image} \\
(5 \times 5)
\end{array}, \begin{array}{c}
\text{kernel} \\
(2 \times 2)
\end{array} \right) = \begin{array}{c}
\text{output} \\
(? \times ?)
\end{array}
\]
Convolution layer properties - output size

\[
\text{conv} \left( \begin{array}{c}
\text{image (5x5)} \\
\text{kernel (2x2)}
\end{array} \right) = \begin{array}{c}
\text{output (?) x (?)}
\end{array}
\]

\[
\text{output size = (new width) x (new height)}
\]
Convolution layer properties - output size

\[
\text{conv} \left( \begin{array}{c}
\text{image} \\
(5\times5)
\end{array} \right)
, \quad \begin{array}{c}
\text{kernel} \\
(2\times2)
\end{array}
\right) = \begin{array}{c}
\text{output} \\
(4\times4)
\end{array}
\]
Convolution layer properties - stride

stride = 1

kernel moves by 1 pixel

$\text{conv} \left( \begin{array}{c}
\text{image (5x5)} \\
\end{array} \right, \begin{array}{c}
\text{kernel (2x2)} \\
\end{array} \right) = \begin{array}{c}
\text{output (4x4)} \\
\end{array}$
Convolution layer properties - stride

stride = 3

kernel moves by 3 pixels

conv (image (5x5), kernel (2x2)) = output (? x ?)
Convolution layer properties - stride

stride = 3

kernel moves by 3 pixels

\[
\text{conv} \left( \begin{array}{c}
\text{image} \\
(5\times5)
\end{array} \right), \quad \begin{array}{c}
\text{kernel} \\
(2\times2)
\end{array} \right) = \begin{array}{c}
\text{output} \\
(? \times ?)
\end{array}
\]
Convolution layer properties - stride

\[ \text{stride} = 3 \]

kernel moves by 3 pixels

\[
\text{conv} \left( \begin{array}{ccc}
\text{image} & \text{kernel} & \text{output} \\
(5x5) & (2x2) & (?) x (?)
\end{array} \right)
\]
Convolution layer properties - stride

stride = 3

kernel moves by 3 pixels

\[
\text{conv} \left( \begin{array}{c}
\text{image (5x5)} \\
\end{array} \right), \quad \begin{array}{c}
\text{kernel (2x2)} \\
\end{array} \right) = \begin{array}{c}
\text{output (\_ \times \_)} \\
\end{array}
\]
Convolution layer properties - stride

\[ \text{stride} = 3 \]

kernel moves by 3 pixels

\[
\begin{array}{ccc}
\text{image} & \text{kernel} & \text{output} \\
(5x5) & (2x2) & (2x2)
\end{array}
\]
Convolution layer properties - stride

\[ M = \frac{(N-K)}{\text{stride}} + 1 \]

![Diagram of convolution process](image)

- **Image** $(N \times N)$
- **Kernel** $(K \times K)$
- **Output** $(M \times M)$

E.g. \( M = (5-2) / 3 + 1 = 2 \)
Convolution layer properties - pad

pad = 1

\[
\text{image (5x5)} \quad \text{kernel (2x2)} \quad \text{output (6x6)}
\]

\[
\begin{array}{cccccc}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
\end{array}
\]

\[
\begin{array}{cccc}
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
\end{array}
\]

\[
\begin{array}{cccccc}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
\end{array}
\]

\[
\uparrow\downarrow
\]

added border of size 1
Convolution layer properties - pad

\[ M = \frac{(N+2\times\text{pad}-K)}{\text{stride}} + 1 \]

\[ \text{image (N+2\times\text{pad})x(N+2\times\text{pad})} \quad , \quad \text{kernel (KxK)} \quad \rightarrow \quad \text{output (MxM)} \]
Multi-channel convolution

\[
\text{conv} \left( \begin{array}{c}
\text{RGB image} \\
(5 \times 5 \times 3)
\end{array} \right), \quad \begin{array}{c}
\text{kernel} \\
(2 \times 2 \times 3)
\end{array} \right) = \begin{array}{c}
\text{output} \\
(4 \times 4 \times 1)
\end{array}
\]
Multi-channel convolution

\[ \text{conv} \left( \begin{array}{c}
\text{RGB image} \\
(5x5x3)
\end{array} \right), \begin{array}{c}
\text{kernel} \\
(2x2x3)
\end{array} \right) = \begin{array}{c}
\text{output} \\
(4x4x1)
\end{array} \]
Multi-channel convolution

\[
\text{conv} \left( \begin{array}{c}
\text{RGB image (5x5x3)}
\end{array} \right)
, \quad \begin{array}{c}
\text{kernel (2x2x3)}
\end{array}
\right) = \begin{array}{c}
\text{output (4x4x1)}
\end{array}
\]
Multi-channel convolution

\[
\text{conv}\left(\begin{array}{c}
\text{RGB image} \\
(5x5x3)
\end{array}\right),
\begin{array}{c}
\text{kernel} \\
(2x2x3)
\end{array}\right) =
\begin{array}{c}
\text{output} \\
(4x4x1)
\end{array}
\]
Multi-channel convolution

\[
\text{conv}\left(\begin{array}{c}
\text{RGB image} \\
(5\times5\times3)
\end{array},
\begin{array}{c}
\text{kernel} \\
(2\times2\times3)
\end{array}\right) =
\begin{array}{c}
\text{output} \\
(4\times4\times1)
\end{array}
\]
Convolutional network (ConvNet)

5x5x3 feature map

layer: conv1

4x4x3 feature map

layer: sigmoid

4x4x3 feature map

layer: conv2

3x3x4 feature map
Convolution backward pass

Learning of convolutional neuron => backpropagation
Convolution backward pass

\[
\begin{array}{c|c}
\frac{\partial p}{\partial w_{11}} & \frac{\partial p}{\partial w_{12}} \\
\frac{\partial p}{\partial w_{21}} & \frac{\partial p}{\partial w_{22}} \\
\end{array}
= ?
\]

\[
\begin{array}{c|c}
w_{11} & w_{12} \\
w_{21} & w_{22} \\
\end{array}
\]

\[
\begin{array}{c|c|c}
x_{11} & x_{12} & x_{13} \\
x_{21} & x_{22} & x_{23} \\
x_{31} & x_{32} & x_{33} \\
\end{array}
\]

\[
\text{conv}(x, w)
\]

\[
\begin{array}{c|c}
y_{11} & y_{12} \\
y_{21} & y_{22} \\
\end{array}
\]

\[
p
\]

Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Cybernetics
Convolution backward pass

\[
\frac{\partial p}{\partial w_{11}} = ?
\]

\(\begin{array}{c|c|c}
 w_{11} & w_{12} \\
 w_{21} & w_{22} \\
\end{array}\)

\(\begin{array}{c}
x_{11} & x_{12} & x_{13} \\
x_{21} & x_{22} & x_{23} \\
x_{31} & x_{32} & x_{33} \\
\end{array}\)

\(\begin{array}{c|c|c|c|c}
 \frac{\partial p}{\partial y_{11}} & \frac{\partial p}{\partial y_{12}} \\
 \frac{\partial p}{\partial y_{21}} & \frac{\partial p}{\partial y_{22}} \\
\end{array}\)

upstream gradient

\(\begin{array}{c|c|c|c|c}
 y_{11} & y_{12} & y_{13} \\
y_{21} & y_{22} & y_{23} \\
y_{31} & y_{32} & y_{33} \\
\end{array}\)

\(\begin{array}{c}
p \\
\end{array}\)
Convolution backward pass

\[
\frac{\partial p}{\partial w_{11}} = \frac{\partial p}{\partial y_{11}} \frac{\partial y_{11}}{\partial w_{11}} + \frac{\partial p}{\partial y_{12}} \frac{\partial y_{12}}{\partial w_{11}} + \frac{\partial p}{\partial y_{21}} \frac{\partial y_{21}}{\partial w_{11}} + \frac{\partial p}{\partial y_{22}} \frac{\partial y_{22}}{\partial w_{11}}
\]
Convolution backward pass

\[
\frac{\partial p}{\partial w_{11}} = \frac{\partial p}{\partial y_{11}} \frac{\partial y_{11}}{\partial w_{11}} + \frac{\partial p}{\partial y_{12}} \frac{\partial y_{12}}{\partial w_{11}} + \frac{\partial p}{\partial y_{21}} \frac{\partial y_{21}}{\partial w_{11}} + \frac{\partial p}{\partial y_{22}} \frac{\partial y_{22}}{\partial w_{11}}
\]

\[
\frac{\partial y_{11}}{\partial w_{11}} = \frac{\partial (w_{11}x_{11} + w_{12}x_{12} + w_{21}x_{21} + w_{22}x_{22})}{\partial w_{11}} = x_{11}
\]
**Convolution backward pass**

\[
\frac{\partial p}{\partial w_{11}} = \frac{\partial p}{\partial y_{11}} x_{11} + \frac{\partial p}{\partial y_{12}} x_{12} + \frac{\partial p}{\partial y_{21}} x_{21} + \frac{\partial p}{\partial y_{22}} x_{22}
\]
Convolution backward pass

\[
\frac{\partial p}{\partial w_{11}} = \frac{\partial p}{\partial y_{11}} x_{11} + \frac{\partial p}{\partial y_{12}} x_{12} + \frac{\partial p}{\partial y_{21}} x_{21} + \frac{\partial p}{\partial y_{22}} x_{22} \\
\frac{\partial p}{\partial w_{12}} = \frac{\partial p}{\partial y_{11}} x_{12} + \frac{\partial p}{\partial y_{12}} x_{13} + \frac{\partial p}{\partial y_{21}} x_{22} + \frac{\partial p}{\partial y_{22}} x_{23}
\]

\[
\begin{array}{cc}
w_{11} & w_{12} \\
w_{21} & w_{22}
\end{array}
\]

\[
\begin{array}{cc}
x_{11} & x_{12} & x_{13} \\
x_{21} & x_{22} & x_{23} \\
x_{31} & x_{32} & x_{33}
\end{array}
\]

upstream gradient

\[
\begin{array}{cc}
\frac{\partial p}{\partial y_{11}} & \frac{\partial p}{\partial y_{12}} \\
\frac{\partial p}{\partial y_{21}} & \frac{\partial p}{\partial y_{22}}
\end{array}
\]

\[
\begin{array}{cc}
y_{11} & y_{12} \\
y_{21} & y_{22}
\end{array}
\]

\[
\begin{array}{c}
\text{conv}(x, w) \\
\text{...}
\end{array}
\]

\[
p
\]
Convolution backward pass

\[
\frac{\partial p}{\partial w_{11}} = \frac{\partial p}{\partial y_{11}} x_{11} + \frac{\partial p}{\partial y_{12}} x_{12} + \frac{\partial p}{\partial y_{21}} x_{21} + \frac{\partial p}{\partial y_{22}} x_{22} \\
\frac{\partial p}{\partial w_{12}} = \frac{\partial p}{\partial y_{11}} x_{12} + \frac{\partial p}{\partial y_{12}} x_{13} + \frac{\partial p}{\partial y_{21}} x_{22} + \frac{\partial p}{\partial y_{22}} x_{23} \\
\frac{\partial p}{\partial w_{21}} = \frac{\partial p}{\partial y_{11}} x_{21} + \frac{\partial p}{\partial y_{12}} x_{22} + \frac{\partial p}{\partial y_{21}} x_{31} + \frac{\partial p}{\partial y_{22}} x_{32}
\]
Convolution backward pass

\[
\begin{align*}
\frac{\partial p}{\partial w_{11}} &= \frac{\partial p}{\partial y_{11}} x_{11} + \frac{\partial p}{\partial y_{12}} x_{12} + \frac{\partial p}{\partial y_{21}} x_{21} + \frac{\partial p}{\partial y_{22}} x_{22} \\
\frac{\partial p}{\partial w_{12}} &= \frac{\partial p}{\partial y_{11}} x_{12} + \frac{\partial p}{\partial y_{12}} x_{13} + \frac{\partial p}{\partial y_{21}} x_{22} + \frac{\partial p}{\partial y_{22}} x_{23} \\
\frac{\partial p}{\partial w_{21}} &= \frac{\partial p}{\partial y_{11}} x_{21} + \frac{\partial p}{\partial y_{12}} x_{22} + \frac{\partial p}{\partial y_{21}} x_{31} + \frac{\partial p}{\partial y_{22}} x_{32} \\
\frac{\partial p}{\partial w_{22}} &= \frac{\partial p}{\partial y_{11}} x_{22} + \frac{\partial p}{\partial y_{12}} x_{23} + \frac{\partial p}{\partial y_{21}} x_{32} + \frac{\partial p}{\partial y_{22}} x_{33}
\end{align*}
\]

$$\text{conv}(x, w)$$

upstream gradient

Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Cybernetics
Convolution backward pass

\[
\frac{\partial p}{\partial w_{11}} = \frac{\partial p}{\partial y_{11}} x_{11} + \frac{\partial p}{\partial y_{12}} x_{12} + \frac{\partial p}{\partial y_{21}} x_{21} + \frac{\partial p}{\partial y_{22}} x_{22} \\
\frac{\partial p}{\partial w_{12}} = \frac{\partial p}{\partial y_{11}} x_{11} + \frac{\partial p}{\partial y_{12}} x_{13} + \frac{\partial p}{\partial y_{21}} x_{22} + \frac{\partial p}{\partial y_{22}} x_{23} \\
\frac{\partial p}{\partial w_{21}} = \frac{\partial p}{\partial y_{11}} x_{21} + \frac{\partial p}{\partial y_{12}} x_{22} + \frac{\partial p}{\partial y_{21}} x_{31} + \frac{\partial p}{\partial y_{22}} x_{32} \\
\frac{\partial p}{\partial w_{22}} = \frac{\partial p}{\partial y_{11}} x_{22} + \frac{\partial p}{\partial y_{12}} x_{23} + \frac{\partial p}{\partial y_{21}} x_{32} + \frac{\partial p}{\partial y_{22}} x_{33}
\]

\( \text{conv}(x, w) \)

Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Cybernetics
### Convolution backward pass

\[
\frac{\partial p}{\partial w_{11}} = \frac{\partial p}{\partial y_{11}} x_{11} + \frac{\partial p}{\partial y_{12}} x_{12} + \frac{\partial p}{\partial y_{21}} x_{21} + \frac{\partial p}{\partial y_{22}} x_{22}
\]

\[
\frac{\partial p}{\partial w_{12}} = \frac{\partial p}{\partial y_{11}} x_{12} + \frac{\partial p}{\partial y_{12}} x_{13} + \frac{\partial p}{\partial y_{21}} x_{22} + \frac{\partial p}{\partial y_{22}} x_{23}
\]

\[
\frac{\partial p}{\partial w_{21}} = \frac{\partial p}{\partial y_{11}} x_{21} + \frac{\partial p}{\partial y_{12}} x_{22} + \frac{\partial p}{\partial y_{21}} x_{31} + \frac{\partial p}{\partial y_{22}} x_{32}
\]

\[
\frac{\partial p}{\partial w_{22}} = \frac{\partial p}{\partial y_{11}} x_{22} + \frac{\partial p}{\partial y_{12}} x_{23} + \frac{\partial p}{\partial y_{21}} x_{32} + \frac{\partial p}{\partial y_{22}} x_{33}
\]

\[
\begin{array}{|c|c|c|}
\hline
\frac{\partial p}{\partial w_{11}} & \frac{\partial p}{\partial w_{12}} & \frac{\partial p}{\partial w_{21}} & \frac{\partial p}{\partial w_{22}} \\
\hline
\end{array}
\quad = \quad \text{conv} \quad \begin{pmatrix}
\begin{array}{ccc}
 x_{11} & x_{12} & x_{13} \\
 x_{21} & x_{22} & x_{23} \\
 x_{31} & x_{32} & x_{33}
\end{array}
\end{pmatrix}
\quad \begin{array}{|c|c|}
\hline
\frac{\partial p}{\partial y_{11}} & \frac{\partial p}{\partial y_{12}} \\
\hline
\frac{\partial p}{\partial y_{21}} & \frac{\partial p}{\partial y_{22}} \\
\hline
\end{array}
\]

---

Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Cybernetics
Convolution backward pass

\[
\frac{\partial p}{\partial w_{11}} = \frac{\partial p}{\partial y_{11}} x_{11} + \frac{\partial p}{\partial y_{12}} x_{12} + \frac{\partial p}{\partial y_{21}} x_{21} + \frac{\partial p}{\partial y_{22}} x_{22}
\]

\[
\frac{\partial p}{\partial w_{12}} = \frac{\partial p}{\partial y_{11}} x_{12} + \frac{\partial p}{\partial y_{12}} x_{13} + \frac{\partial p}{\partial y_{21}} x_{22} + \frac{\partial p}{\partial y_{22}} x_{23}
\]

\[
\frac{\partial p}{\partial w_{21}} = \frac{\partial p}{\partial y_{11}} x_{21} + \frac{\partial p}{\partial y_{12}} x_{22} + \frac{\partial p}{\partial y_{21}} x_{31} + \frac{\partial p}{\partial y_{22}} x_{32}
\]

\[
\frac{\partial p}{\partial w_{22}} = \frac{\partial p}{\partial y_{11}} x_{22} + \frac{\partial p}{\partial y_{12}} x_{23} + \frac{\partial p}{\partial y_{21}} x_{32} + \frac{\partial p}{\partial y_{22}} x_{33}
\]
Convolution backward pass

\[
\begin{align*}
\frac{\partial p}{\partial w_{11}} &= \frac{\partial p}{\partial y_{11}} x_{11} + \frac{\partial p}{\partial y_{12}} x_{12} + \frac{\partial p}{\partial y_{21}} x_{21} + \frac{\partial p}{\partial y_{22}} x_{22} \\
\frac{\partial p}{\partial w_{12}} &= \frac{\partial p}{\partial y_{11}} x_{12} + \frac{\partial p}{\partial y_{12}} x_{13} + \frac{\partial p}{\partial y_{21}} x_{22} + \frac{\partial p}{\partial y_{22}} x_{23} \\
\frac{\partial p}{\partial w_{21}} &= \frac{\partial p}{\partial y_{11}} x_{21} + \frac{\partial p}{\partial y_{12}} x_{22} + \frac{\partial p}{\partial y_{21}} x_{31} + \frac{\partial p}{\partial y_{22}} x_{32} \\
\frac{\partial p}{\partial w_{22}} &= \frac{\partial p}{\partial y_{11}} x_{22} + \frac{\partial p}{\partial y_{12}} x_{23} + \frac{\partial p}{\partial y_{21}} x_{32} + \frac{\partial p}{\partial y_{22}} x_{33}
\end{align*}
\]

\[\quad = \text{conv} \left( \begin{array}{ccc}
  \frac{\partial p}{\partial y_{11}} & \frac{\partial p}{\partial y_{12}} & \frac{\partial p}{\partial y_{21}} & \frac{\partial p}{\partial y_{22}} \\
  x_{11} & x_{12} & x_{13} & \frac{\partial p}{\partial y_{11}} \\
  x_{21} & x_{22} & x_{23} & \frac{\partial p}{\partial y_{12}} \\
  x_{31} & x_{32} & x_{33} & \frac{\partial p}{\partial y_{21}} \\
  \frac{\partial p}{\partial y_{22}} & \frac{\partial p}{\partial y_{22}} & \frac{\partial p}{\partial y_{22}} & \frac{\partial p}{\partial y_{22}}
\end{array} \right)\]
Convolution backward pass

\[
\frac{\partial p}{\partial w_{11}} = \frac{\partial p}{\partial y_{11}} x_{11} + \frac{\partial p}{\partial y_{12}} x_{12} + \frac{\partial p}{\partial y_{21}} x_{21} + \frac{\partial p}{\partial y_{22}} x_{22} \\
\frac{\partial p}{\partial w_{12}} = \frac{\partial p}{\partial y_{11}} x_{12} + \frac{\partial p}{\partial y_{12}} x_{13} + \frac{\partial p}{\partial y_{21}} x_{22} + \frac{\partial p}{\partial y_{22}} x_{23} \\
\frac{\partial p}{\partial w_{21}} = \frac{\partial p}{\partial y_{11}} x_{21} + \frac{\partial p}{\partial y_{12}} x_{22} + \frac{\partial p}{\partial y_{21}} x_{31} + \frac{\partial p}{\partial y_{22}} x_{32} \\
\frac{\partial p}{\partial w_{22}} = \frac{\partial p}{\partial y_{11}} x_{22} + \frac{\partial p}{\partial y_{12}} x_{23} + \frac{\partial p}{\partial y_{21}} x_{32} + \frac{\partial p}{\partial y_{22}} x_{33}
\]

\[
\text{conv}(\begin{array}{ccc}
\frac{\partial p}{\partial w_{11}} & \frac{\partial p}{\partial w_{12}} \\
\frac{\partial p}{\partial w_{21}} & \frac{\partial p}{\partial w_{22}} \\
\end{array}) = \begin{array}{ccc}
x_{11} & x_{12} & x_{13} \\
x_{21} & x_{22} & x_{23} \\
x_{31} & x_{32} & x_{33} \\
\end{array}
\]

Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Cybernetics
Convolution backward pass wrt weights

- Backpropagation in convolutional layer wrt weights is: “convolution of input feature map with upstream gradient”.

\[
\begin{array}{c|c|c}
\frac{\partial p}{\partial w_{11}} & \frac{\partial p}{\partial w_{12}} \\
\frac{\partial p}{\partial w_{21}} & \frac{\partial p}{\partial w_{22}} \\
\end{array}
\]

\[
\begin{array}{c|c|c|c|c}
\frac{\partial p}{\partial y_{11}} & \frac{\partial p}{\partial y_{12}} \\
\frac{\partial p}{\partial y_{21}} & \frac{\partial p}{\partial y_{22}} \\
\end{array}
\]

\[
\begin{array}{c|c|c|c}
x_{11} & x_{12} & x_{13} \\
x_{21} & x_{22} & x_{23} \\
x_{31} & x_{32} & x_{33} \\
\end{array}
\]

\[
\begin{array}{c|c|c}
y_{11} & y_{12} \\
y_{21} & y_{22} \\
\end{array}
\]

upstream gradient
Convolution backward pass wrt input feature map

Backpropagation in convolutional layer is: "convolution of padded upstream gradient with mirrored weights"

\[
\begin{array}{ccc}
\frac{\partial p}{\partial x_{11}} & \frac{\partial p}{\partial x_{12}} & \frac{\partial p}{\partial x_{13}} \\
\frac{\partial p}{\partial x_{21}} & \frac{\partial p}{\partial x_{22}} & \frac{\partial p}{\partial x_{23}} \\
\frac{\partial p}{\partial x_{31}} & \frac{\partial p}{\partial x_{32}} & \frac{\partial p}{\partial x_{33}}
\end{array}
\]

\[
\begin{array}{cccc}
0 & 0 & 0 & 0 \\
0 & \frac{\partial p}{\partial y_{11}} & \frac{\partial p}{\partial y_{12}} & 0 \\
0 & \frac{\partial p}{\partial y_{21}} & \frac{\partial p}{\partial y_{22}} & 0 \\
0 & 0 & 0 & 0
\end{array}
\]

\[
\text{conv}\left( w_{22} w_{21} w_{12} w_{11} \right)
\]

\[
\text{conv}\left( x, w \right)
\]

upstream gradient
 Convolutional network backprop
Convolutional net

- Convolutional network (ConvNet) is concatenation of convolutional layers
- Backprop in ConvNet is convolution of feature maps or kernels with upstream gradient.
- Feed-forward and backprop are convolutions => efficient implementation on GPU
LeCun’s letter recognition 1998 (over 13k citations !!!)

LeCun et al, Gradient based learning applied to document recognition, IEEE, 1998
Classification results

http://image-net.org/challenges/LSVRC/2017/index

Steel drum

\[
\text{Error} = \frac{1}{100,000} \sum_{i=1}^{100,000} 1[\text{incorrect on image } i]
\]
Classification results

AlexNet
8 layers

Classification Error

<table>
<thead>
<tr>
<th>Year</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>2010</td>
<td>0.28</td>
</tr>
<tr>
<td>2011</td>
<td>0.26</td>
</tr>
<tr>
<td>2012</td>
<td>0.28</td>
</tr>
</tbody>
</table>
Classification results

**AlexNet**
- 8 layers
- 2010: 0.28
- 2011: 0.26
- 2012: 0.16
- 2013: 0.12

**VGGnet**
- 19 layers

Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Cybernetics
Classification results

AlexNet
8 layers
VGGnet
19 layers
GoogLeNet
22 layers
Classification results

**AlexNet**
8 layers
Classification Error: 0.28

**VGGnet**
19 layers
Classification Error: 0.26

**GoogLeNet**
22 layers
Classification Error: 0.16

**ResNet**
152 layers
Classification Error: 0.07

Classification Error: 0.036

Year:
- 2010
- 2011
- 2012
- 2013
- 2014
- 2015
Classification results

AlexNet
8 layers
2010: 0.28
2011: 0.26

VGGNet
19 layers
2012: 0.16

GoogLeNet
22 layers
2013: 0.12

ResNet
152 layers
2014: 0.07
2015: 0.036
2016: 0.03
2017: 0.023

16.7% ↓ 23.3% ↓
Demo

- convnet demo from Karpathy: https://cs.stanford.edu/people/karpathy/convnetjs/demo/cifar10.html
Next lecture

• gradient learning (what make it tough)
• other layers:
  • activation function,
  • batch normalization,
  • drop out,
  • loss layers