Hot Research Topics in AI for Engineering Applications

Bonus Content - Convolutional Neural Networks and Regularization in NNs
WS 2025/26

Convolutional Neural Networks

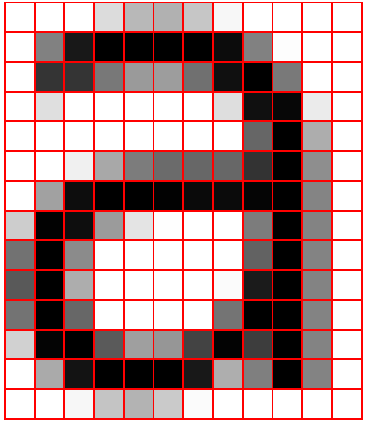
Convolutional Neural Networks

- **Convolutional Neural Networks** (CNNs) are a specific architecture of neural networks especially suited for **image processing** tasks
- Basic Idea
  - Implementation of a receptive field for each neuron
  - Conservation of a spatial context between layers
  - Each Neuron only depends on a number of spatially close neurons from the previous layer

- Ideas originate in the 1980s but where early on hindered by hardware limitations
- CNNs were successfully applied to yield unprecedented results on image analysis tasks in 2012 (AlexNet)

Digital Representation of Images

- Images can be digitally represented by 2-D matrices (greyscale)
  - Sampled into pixels
  - Each pixel has a value
    - High value = white
    - Low value = black
- In case of color images, 3-D matrices are used
  - 2-D spatial dimension
  - 3 color channels: red, green, blue



1.0	1.0	1.0	0.9	0.6	0.6	0.6	1.0	1.0	1.0	1.0	1.0
1.0	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.5	1.0	1.0	1.0
1.0	0.2	0.2	0.5	0.6	0.6	0.5	0.0	0.0	0.5	1.0	1.0
1.0	0.9	1.0	1.0	1.0	1.0	1.0	0.9	0.0	0.0	0.9	1.0
1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.5	0.0	0.5	1.0
1.0	1.0	1.0	0.5	0.5	0.5	0.5	0.5	0.4	0.0	0.5	1.0
1.0	0.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5	1.0
0.9	0.0	0.0	0.6	1.0	1.0	1.0	1.0	0.5	0.0	0.5	1.0
0.5	0.0	0.6	1.0	1.0	1.0	1.0	1.0	0.5	0.0	0.5	1.0
0.5	0.0	0.7	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.5	1.0
0.6	0.0	0.6	1.0	1.0	1.0	1.0	0.5	0.0	0.0	0.5	1.0
0.9	0.1	0.0	0.6	0.7	0.7	0.5	0.0	0.5	0.0	0.5	1.0
1.0	0.7	0.1	0.0	0.0	0.0	0.1	0.9	0.8	0.0	0.5	1.0
1.0	1.0	1.0	0.8	0.8	0.9	1.0	1.0	1.0	1.0	1.0	1.0

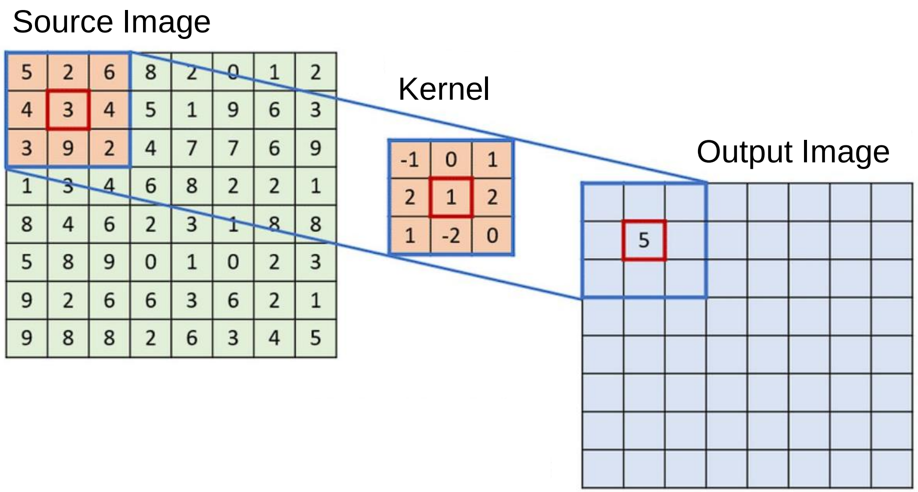
Source: https://pippin.gimp.org/image-processing/chap_dir.html

Convolution in Image Processing

Karlsruhe Institute of Technology

- Convolution is a common local filtering technique in image processing
- A **kernel** of fixed size is applied at every image location

- Application of kernel corresponds to weighted sum of considered pixel values



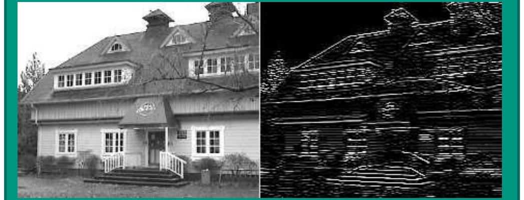
Source: Link

Information Management in Engineering

Convolution in Image Processing

- Different kernels perform different operations

Line Detection



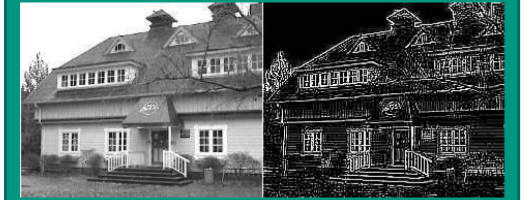
-1	-1	-1
2	2	2
-1	-1	-1

Horizontal Lines

1	2	-1
1	2	-1
1	2	-1

Vertical Lines

Edge Detection



-1	-1	-1
-1	8	-1
-1	-1	-1

0	-1	0
-1	4	-1
0	-1	0

Laplacian

Blur

$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$

Box blur

Gaussian

[illegible]	[illegible]	[illegible]	$\frac{4}{273}$	$\frac{1}{273}$
$\frac{4}{273}$	$\frac{16}{273}$	$\frac{26}{273}$	$\frac{16}{273}$	$\frac{4}{273}$
$\frac{7}{273}$	$\frac{26}{273}$	$\frac{41}{273}$	$\frac{26}{273}$	$\frac{7}{273}$
$\frac{4}{273}$	$\frac{16}{273}$	$\frac{26}{273}$	$\frac{16}{273}$	$\frac{4}{273}$
$\frac{1}{273}$	$\frac{4}{273}$	$\frac{7}{273}$	$\frac{4}{273}$	$\frac{1}{273}$

Source: [Link]

Convolutional Layers in Neural Networks

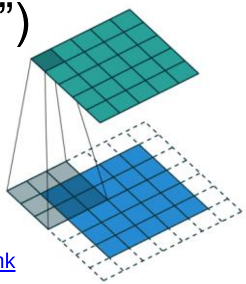
- Neurons in convolutional layers are arranged 3-dimensionally
  - 2 spatial dimensions
  - Additional dimension for multiple channels/filters
- Weighted sum z only considers neurons in spatial vicinity

$$z = \sum_{c=1}^{C} \sum_{i=-m}^{m} \sum_{j=-m}^{m} w_{cij} x_{cij} + b$$

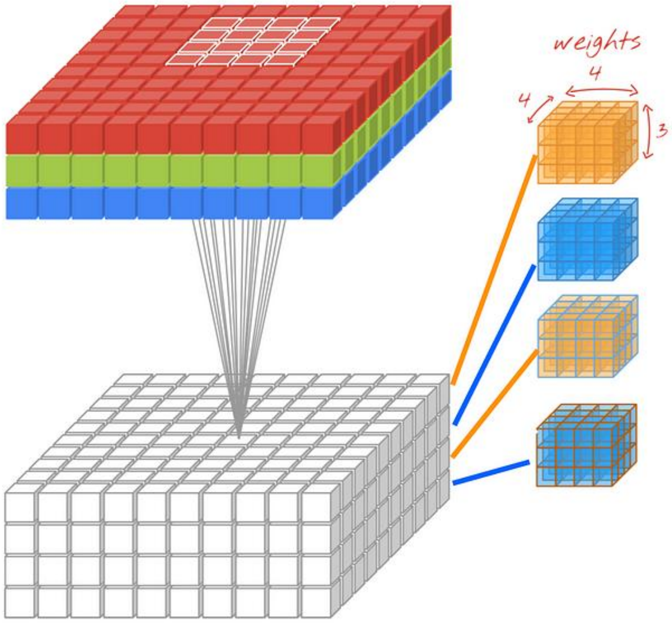
  - Here, c is the depth dimension of the previous layer (e.g. RGB color channels in an input image)
  - The size of the convolution kernel is $2m + 1$
- Activation function applied as before to compute neuron activation a

Convolutional Layers in Neural Networks

- In case of images, the input layer often contains 3 channels
- Spatial arrangement is preserved in convolutional layers
- Each layer contains a number of filters which are stacked in the third dimension
- Handling of boundary pixels needs to be considered, e.g. padding of constant (zero) boundary units ("same padding")



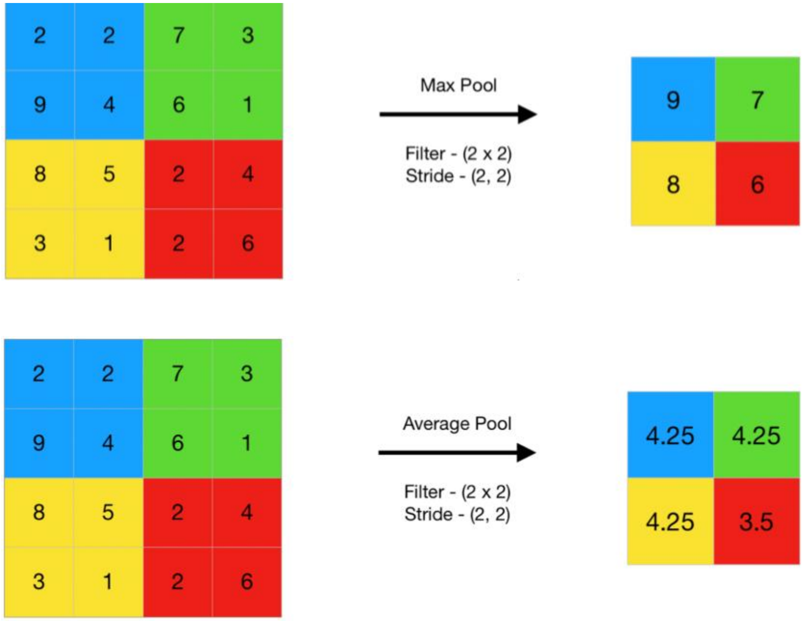
Source: Link



Source: Link

Pooling Layers

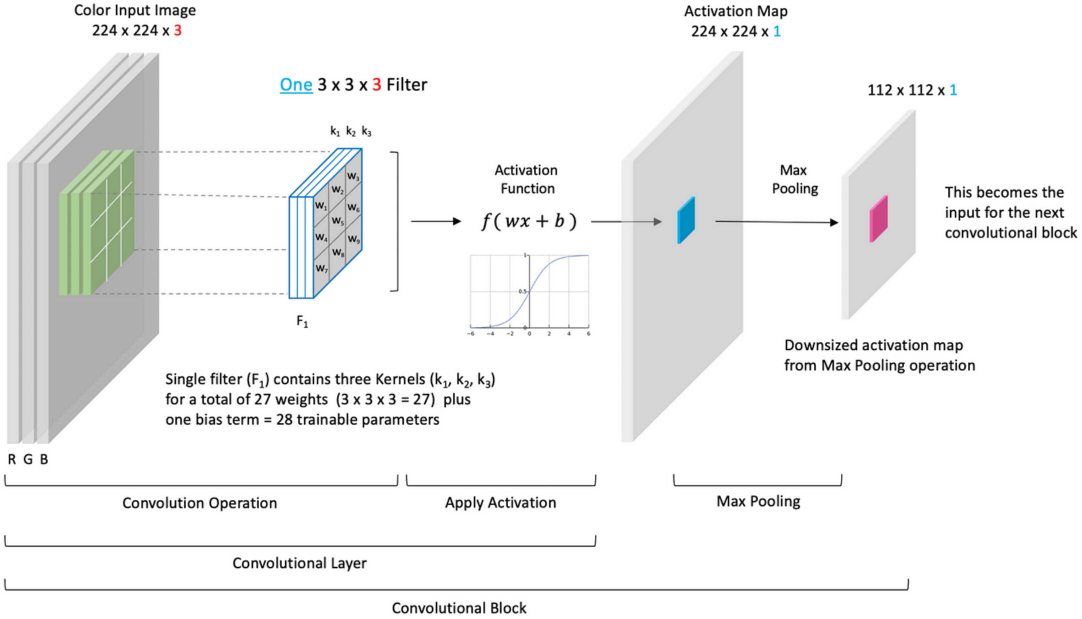
- Used to reduce spatial dimensions of layers in a CNN
- Third (depth) dimension is conserved (applied in each channel independently)
- Different types, most common are
  - Max Pooling
  - Average Pooling
- Parameters:
  - Filter Size – spatial dimension of pooling (usually rectangular window, e.g. 2x2)
  - Stride – distance at which the filter is applied (often equal to filter size)



Source: [Link]

Convolutional Blocks

- In CNNs, convolutional layers and pooling layers are often arranged alternatingly
- Sequence of convolutional layer followed by a pooling layer is often called a **convolutional block**



Source: Link

CNN Architectures

- CNNs originally consist of
  - Convolutional Blocks
  - Fully-connected layers toward the end
  - Softmax classifier at the end
- Conventions
  - Spatial dimensions decrease gradually
  - Channel/filter amount increases gradually

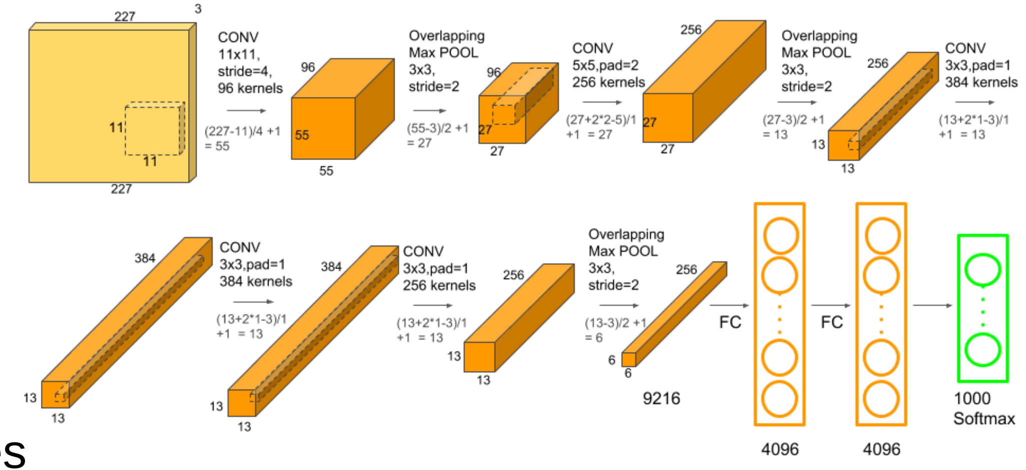


Figure: AlexNet architecture. AlexNet famously won the 2012 ImageNet image classification challenge reviving interest in CNNs for image analysis tasks.

Source: Link

Advanced CNN Architectures

Karlsruhe Institute of Technology

- Newer CNN architecture often use additional concepts
  - Use different convolution kernel sizes in parallel (e.g. Inception)
  - Depth-wise separable convolutions*
  - Skip connections, e.g. residual layers*
  - Transposed convolutions (up-sampling convolutions)*

Information Management in Engineering

Regularization in Neural Networks

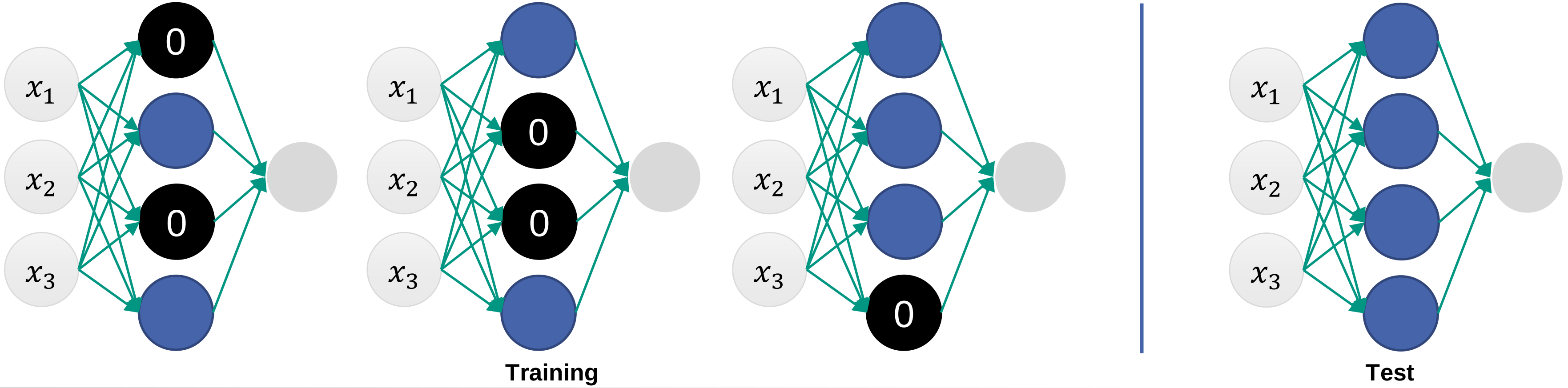
Regularization in Artificial Neural Networks

- L1 or L2 regularization (compare other ML models)
  - Add term to loss function penalizing (large) parameter weights
- Early Stopping
  - Monitor evaluation error during training → stop when it does no longer decrease
- Dropout
  - Random units are set to zero during training
- Batch Normalization[1]
  - During training, normalize layer outputs according to training data specific mean and standard deviation
- Data Augmentation
  - Synthetically increase amount of training data

[1] Ioffe, Sergey. "Batch normalization: Accelerating deep network training by reducing internal covariate shift." *arXiv preprint arXiv:1502.03167* (2015).

Regularization in Artificial Neural Networks: Dropout

- Idea: Prevent feature co-adaptation
- Can be seen as a form of model bagging (training an ensemble model)
- Hyperparameter: Dropout probability p
  - With probability p, each unit is set to 0 during training
  - At test time, all units are used (weights reduced by $1 - p$)



Regularization in Artificial Neural Networks: Batch Normalization

- Additional layer in network architecture
- Normalizes unit outputs according to training data statistics
  - Computes unit value mean μ_B and standard variation σ_B of current batch
  - Keeps a moving average of unit mean $\hat{\mu}$ and standard variation $\hat{\sigma}$ over whole training data
  - Two learnable parameters: target mean β and variance γ

$$BN(x) = \gamma \cdot \frac{x - \hat{\mu}}{\hat{\sigma}} + \beta$$

- Originally applied before computation of activation function
- Can accelerate training, i.e. convergence
- Prevents overfitting to outliers

Regularization in Artificial Neural Networks: Data Augmentation

- Apply label-preserving transformations to input features
- Frequently used for arbitrary types of data
- Efficiently increases amount of training samples (especially important when data acquisition / labelling is costly)

Original

Flipped

Random Saturation

Random Brightness

Rotation

Source: [Link]