Deep Learning Day

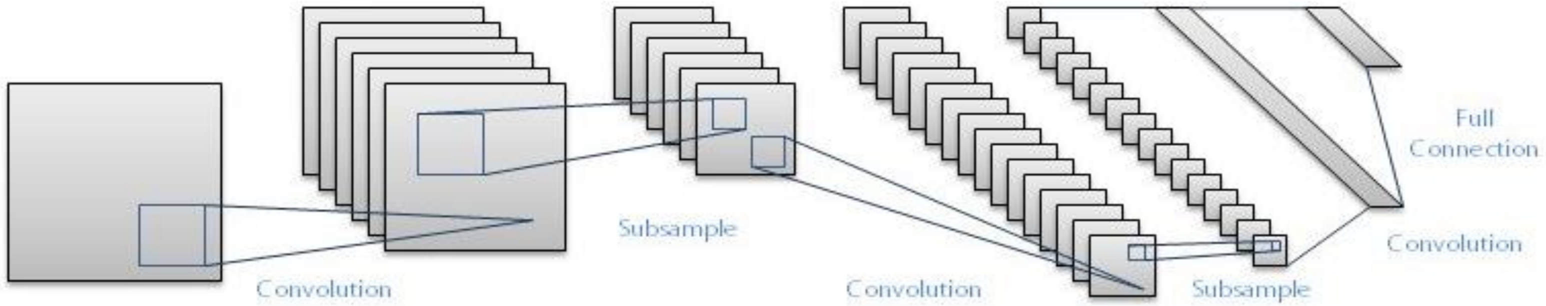
Zurich University
of Applied Sciences

Convolutional Neural Networks (CNNs)

Beate Sick

Institut für Datenanalyse und Prozessdesign

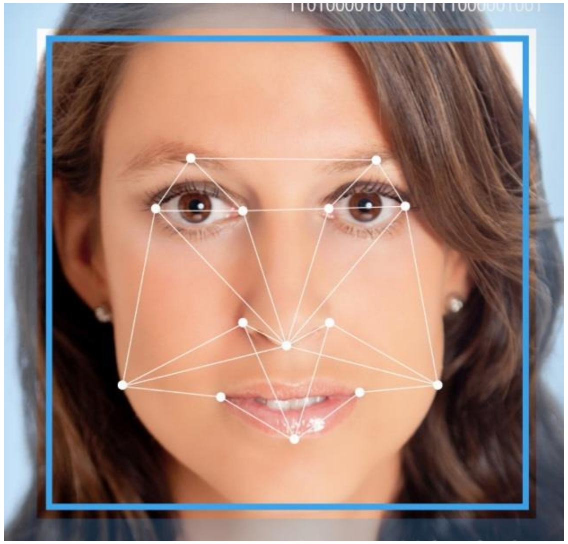
Zürcher Hochschule für Angewandte Wissenschaften



Winterthur, 22th Sep 2017

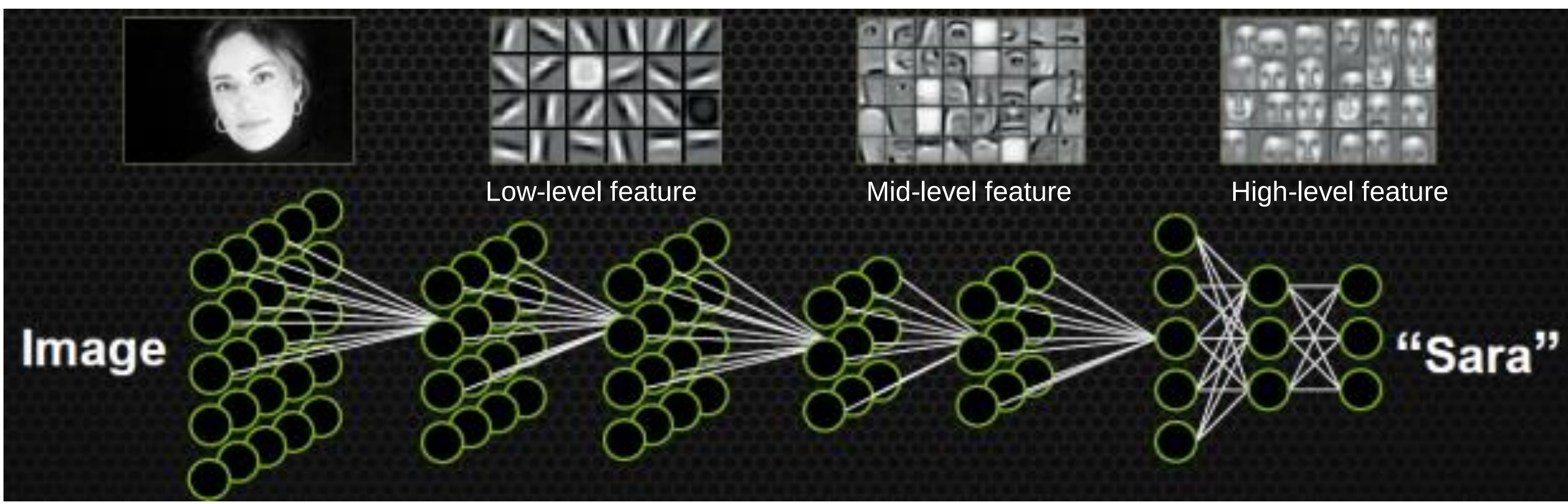
Image credits: http://neuralnetworksanddeeplearning.com/chap6.html, http://www.cemetech.net/projects/ee/scouter/thesis.pdf

What is new in the deep learning approach?

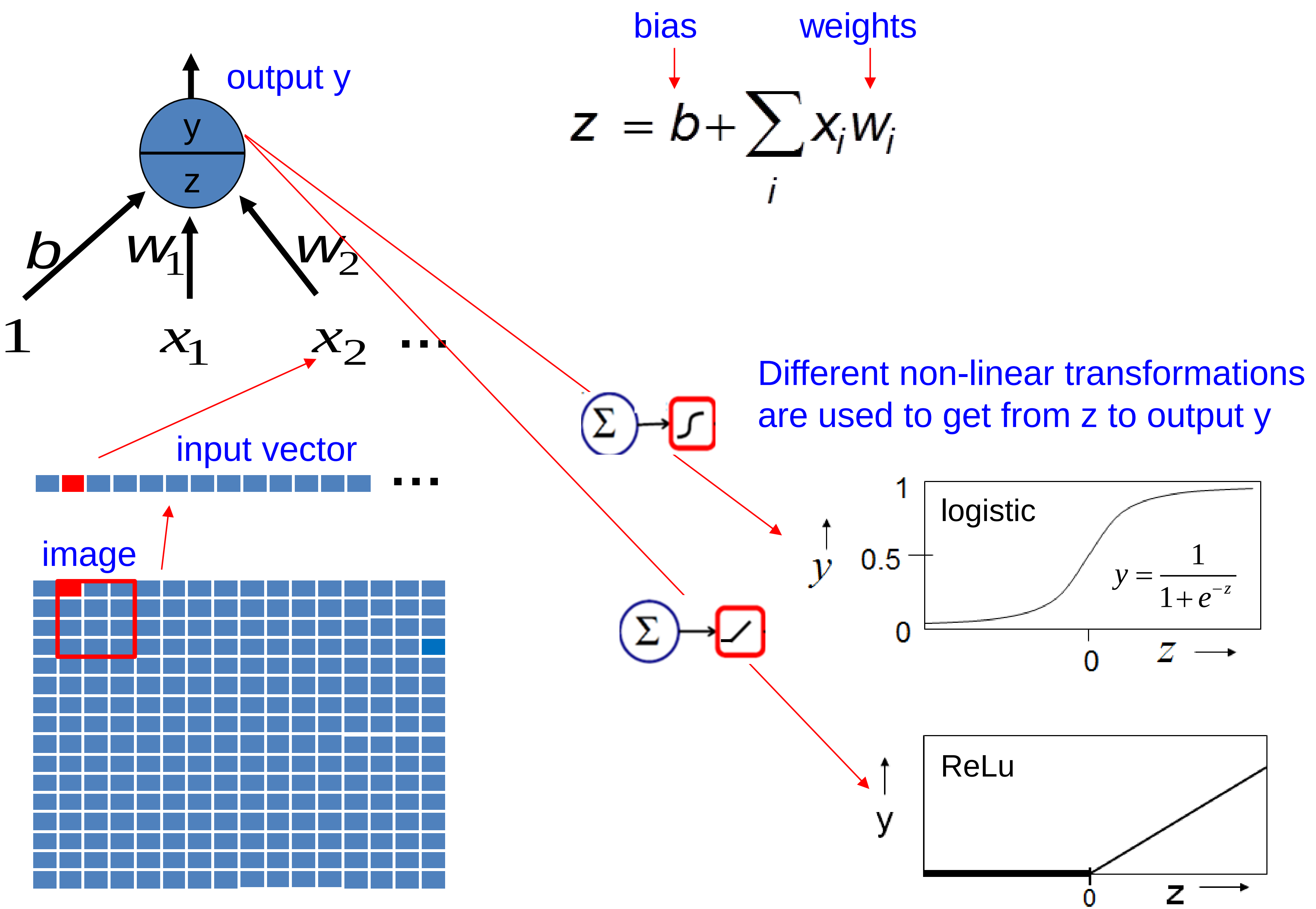


Traditional:
Extract handcrafted features & use these features to train / fit a model (e.g. SVM, RF) and use fitted model to perform classification/prediction.

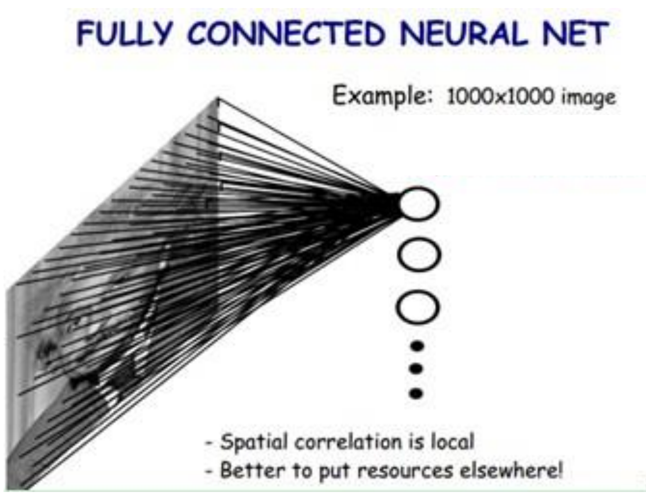
Deep learning:
In deep neural networks start with raw data and learn during training/fitting to extract appropriate hierarchical features and to use them for classification/prediction.

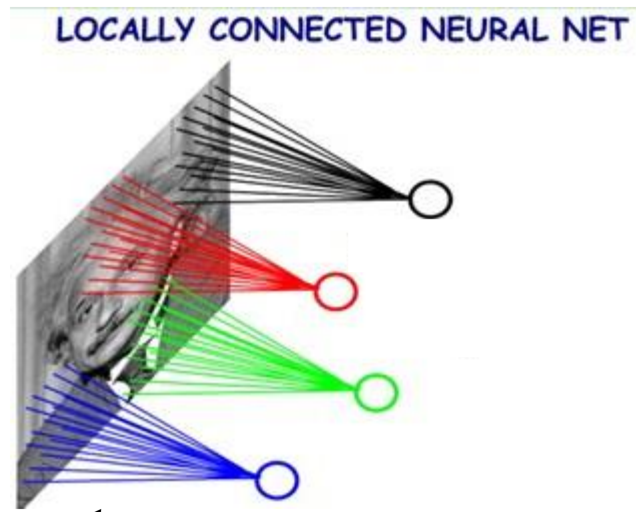


An artificial neuron



Convolution extracts local information using few weights





Shared weights:

by using the same weights for each patch of the image we need much less parameters than in the fully connected NN and get from each patch the same kind of local feature information such as the presence of a edge.

Convolutional networks use neighborhood information and replicated local feature extraction

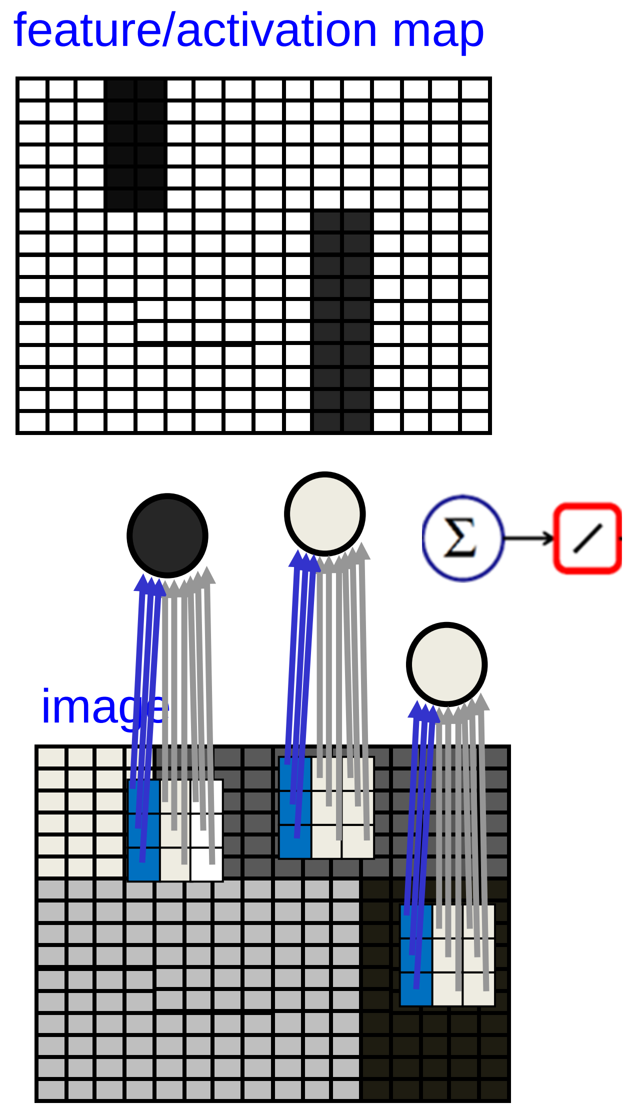
In a locally connected network the calculation rule

$$z = b + \sum_i x_i w_i$$

Pixel values in a small image patch are element-wise multilied with weights of a small filter/kernel:

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

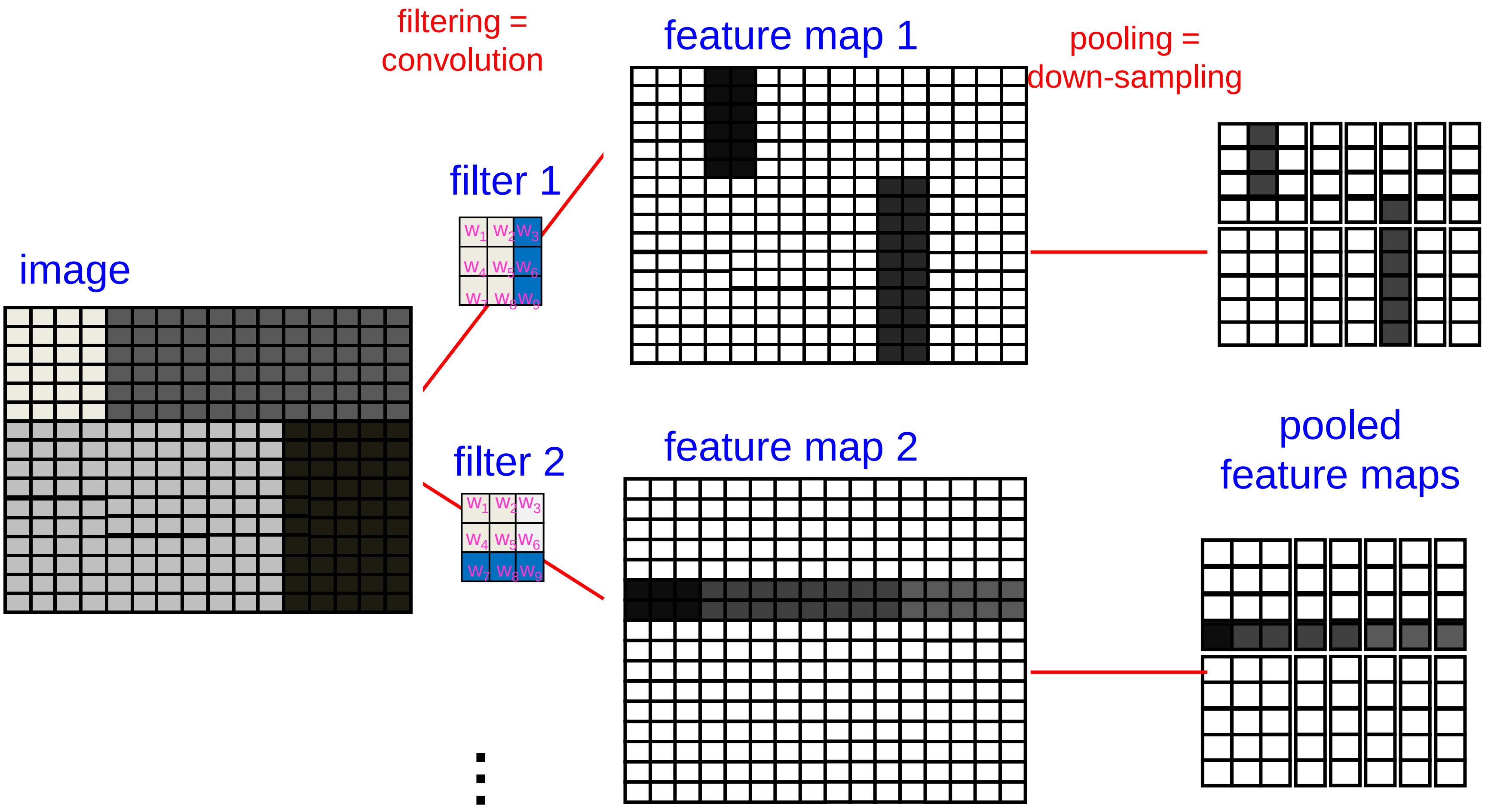
0.9	0.1	0.1
0.9	0.1	0.1
0.9	0.1	0.1



The filter is applied at each position of the image and it can be shown that the result is maximal if the image pattern corresponds to the weight pattern.

The results form again an image called feature map (=activation map) which shows at which position the feature is present.

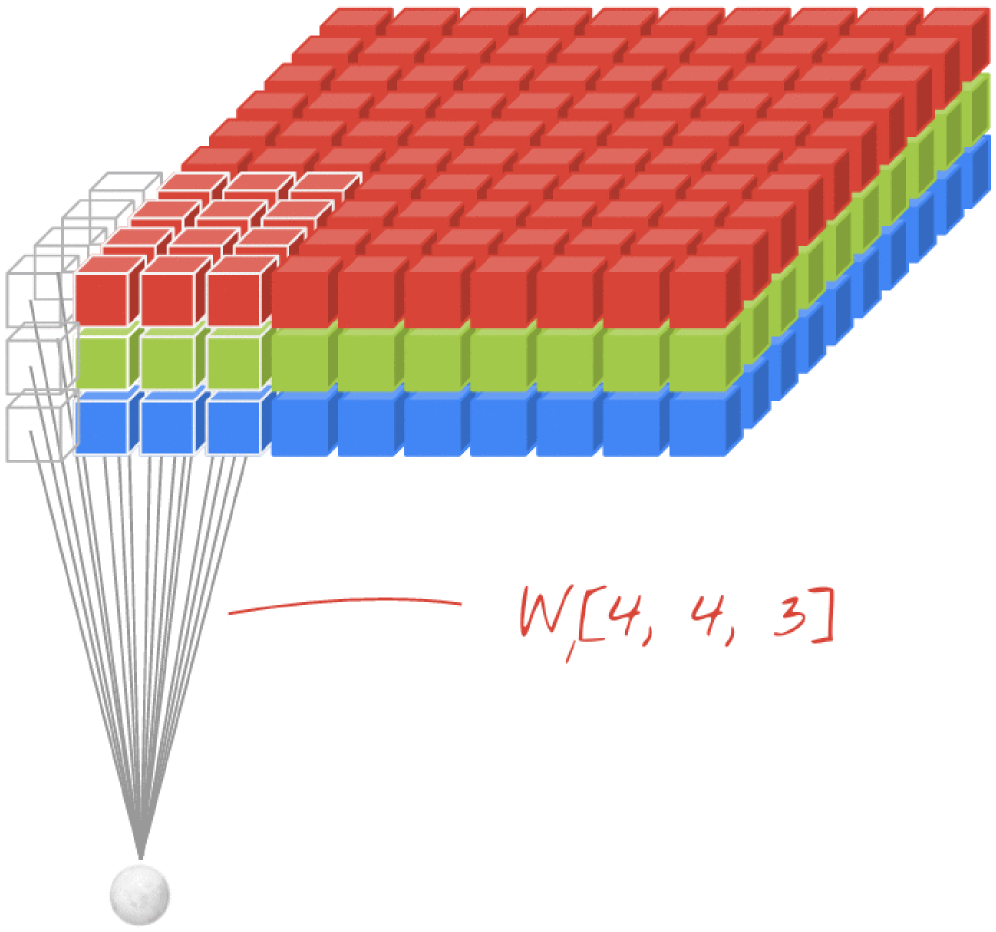
Convolutional networks use neighborhood information and replicated local feature extraction



The weights of each filter are randomly initiated and then adapted during the training.

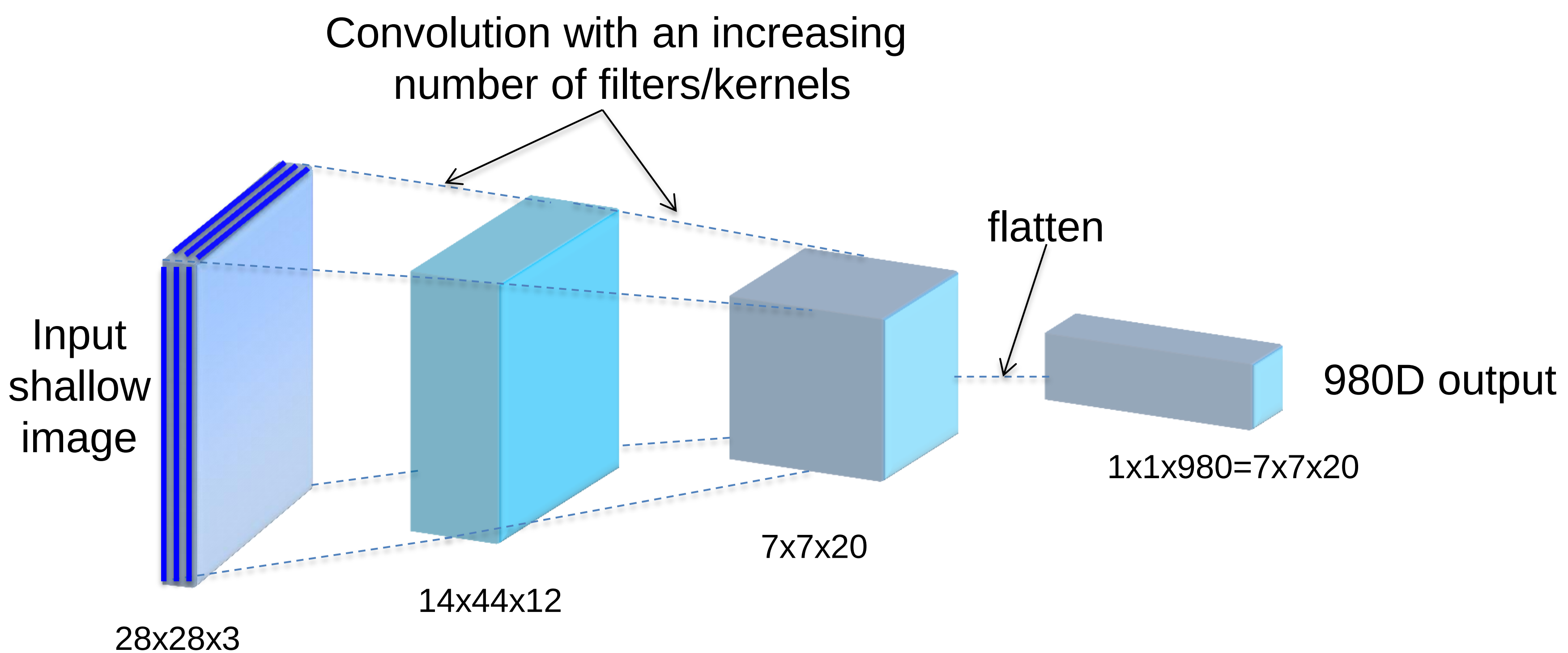
Animated convolution with 3 input channels

3 color channel input image



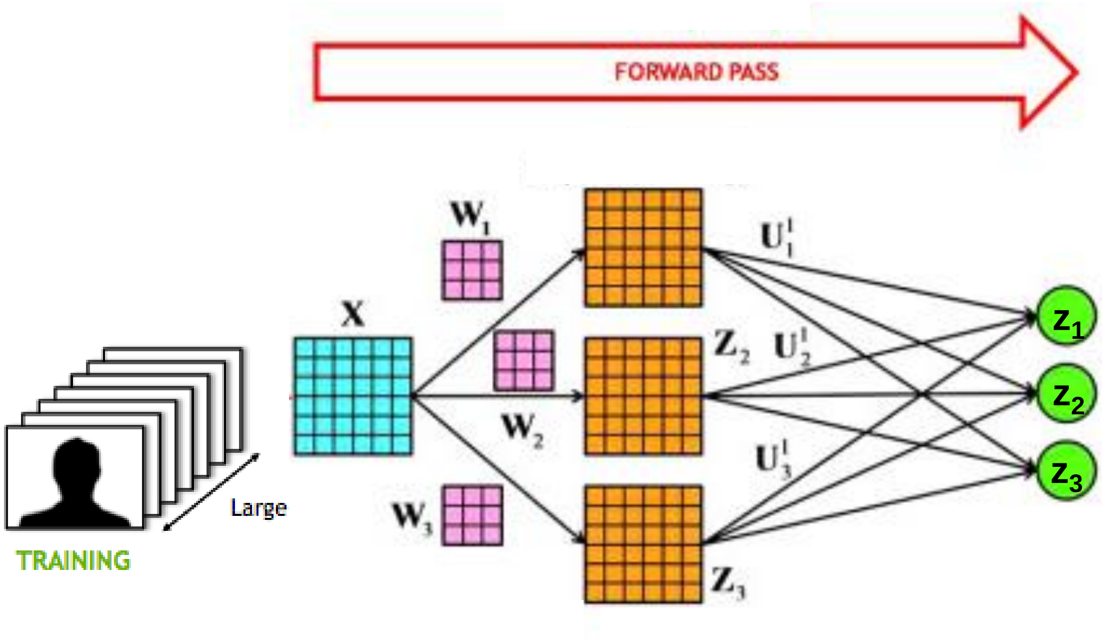
Animation credits: M.Gorner, https://codelabs.developers.google.com/codelabs/cloud-tensorflow-mnist/#10

Typical shape of a classical CNN



Spatial resolution is decreased e.g. via max-pooling while more abstract image features are detected in deeper layers.

A CNN yields for each image a vector of output values



e^{z_1}

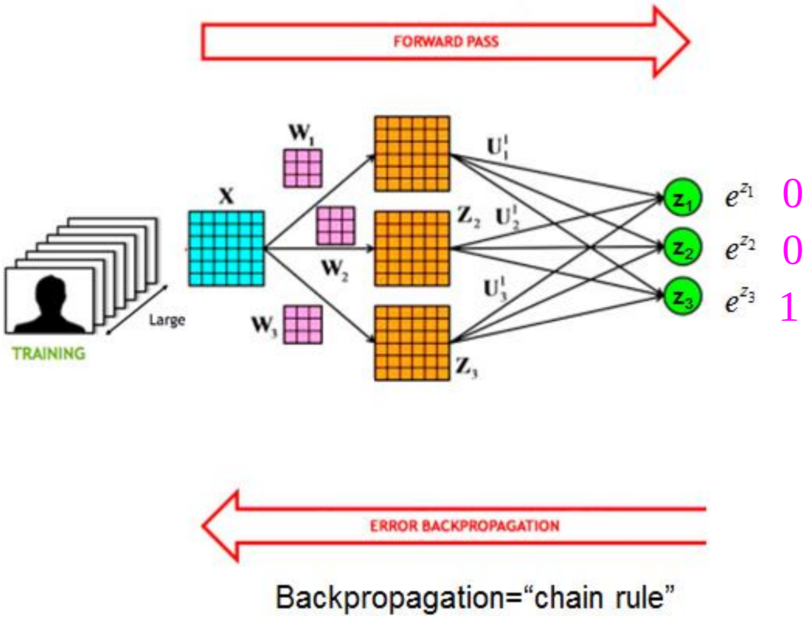
e^{z_2}

e^{z_3}

Output depends on input and on the weights of the CNN.

Image credits: Audhkhasi et al., Neural Networks 78 (2016) 15–23

Training of a CNN is based on gradient backpropagation



For the training we need the **true label** for each image which we then compare with the **output** of the CNN.

We want to adjust the weights in a way so that difference between true label and output is minimal.

Minimize Loss-function:

L=distance(truth, output(w)**)**

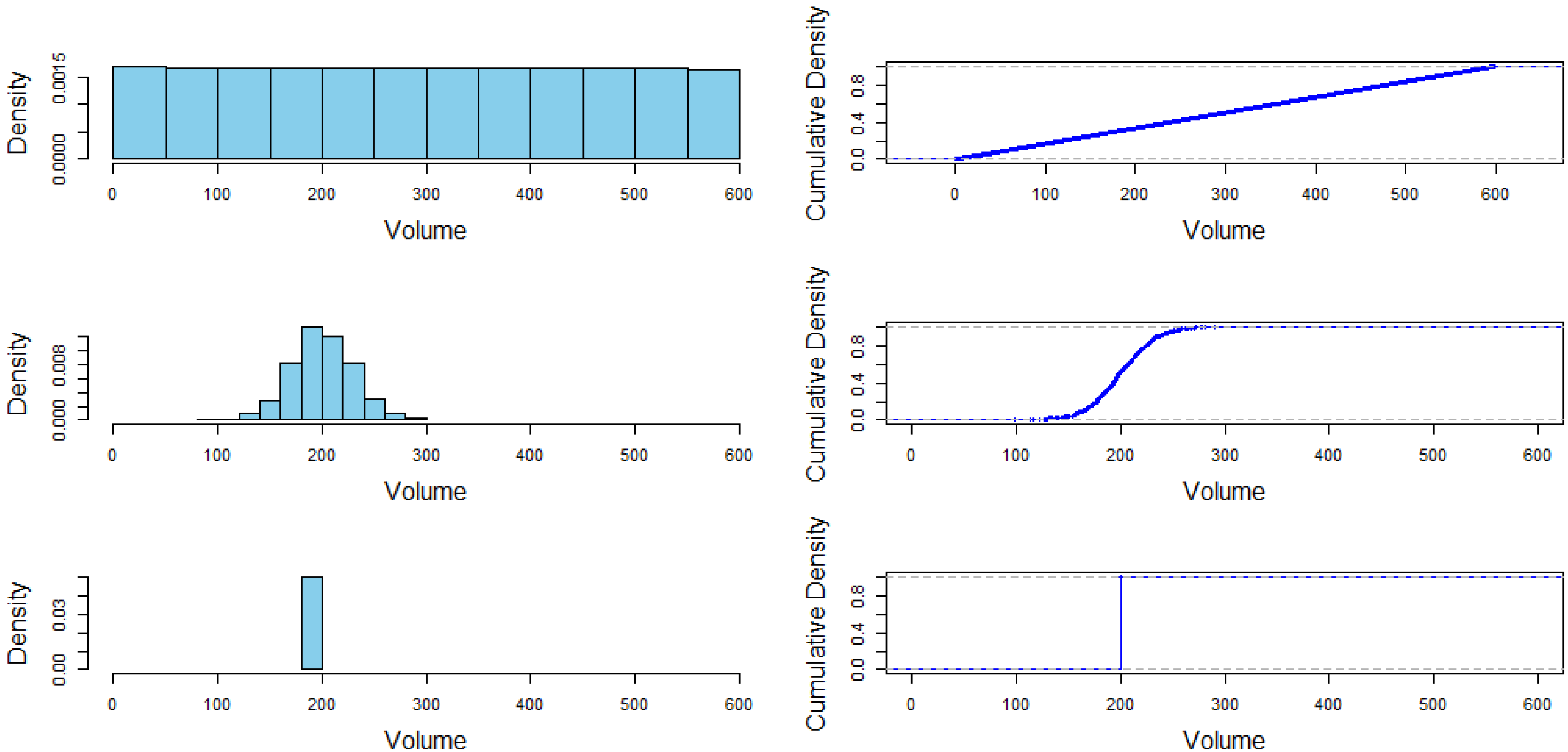
$$w_i^{(t)} = w_i^{(t-1)} - l^{(t)} \left.\frac{\partial L(w)}{\partial w_i}\right|_{w_i = w_i^{(t-1)}}$$

learning rate

Image credits: Audhkhasi et al., Neural Networks 78 (2016) 15–23

Task in tutorial 1

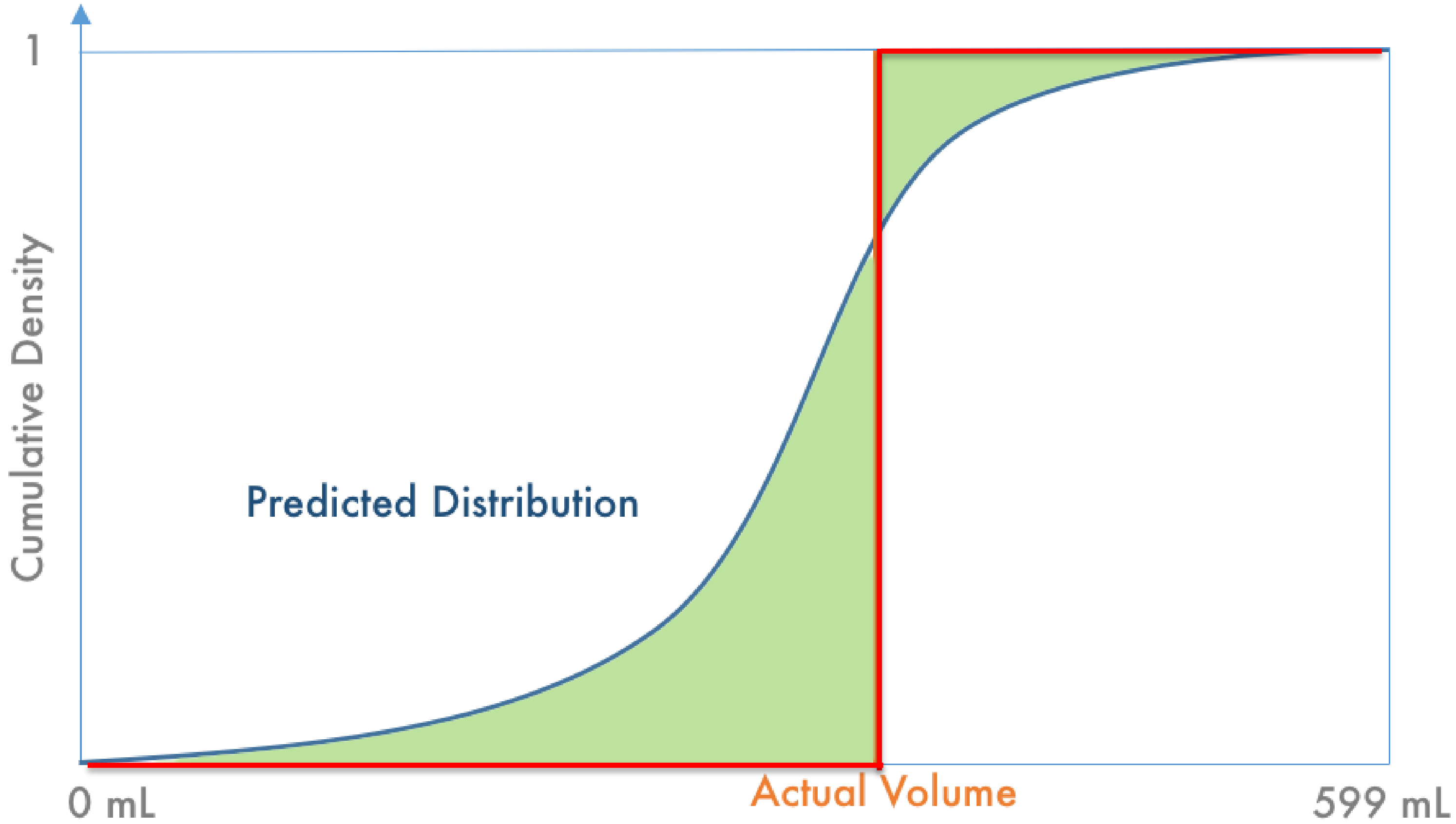
- **Input**: stack of MRI heart images **Output**: plausible heart volumes
- The heart volume can be between 0ml and 599ml.
- Our output layer has 600 neurons – each corresponding to the cumulative probability of each possible Volume $e^{z_i} = P(V \leq Xml)$ with $X \in \{0,1, \ldots 599\}$



Cost or loss function in tutorial 1

Loss-function L: **L = distance(** truth, CNN-output **)**
= green area = sum(|label – pred|)

Note in code the square of the areas is used.



Next

9:15 - 9:35	Oliver and Beate	Welcome and short intro to CNNs
9:30 - 10:15	Kevin Mader (4 Quant)	Approaches and Challegenges for using Artificial Inteligence in Medical Imaging
10:15 - 10:30	Coffee Break	Coffe Break
10:30 - 12:15	Gunter Roth (NVIDIA)	Tutorial: Medical Image Analysis with R and MXNet
12:15 - 13:30	--	Lunch at your own expense (e.g. Mensa ZHAW)
13:10 - 13:30	Oliver	Short intro to RNNs (optional)
13:30 - 14:10	Dirk von Grünigen	Natural Language Dialogues with Sequence-to-Sequence Learning
14:10 - 15:30	Gunter Roth (NVIDIA)	Tutorial: Modelling Time Series Data with Recurrent Neural Networks in Keras
15:30 - 16:00	Various	Spotlight Talks and closing
16:00 -	You	Posters and Apero