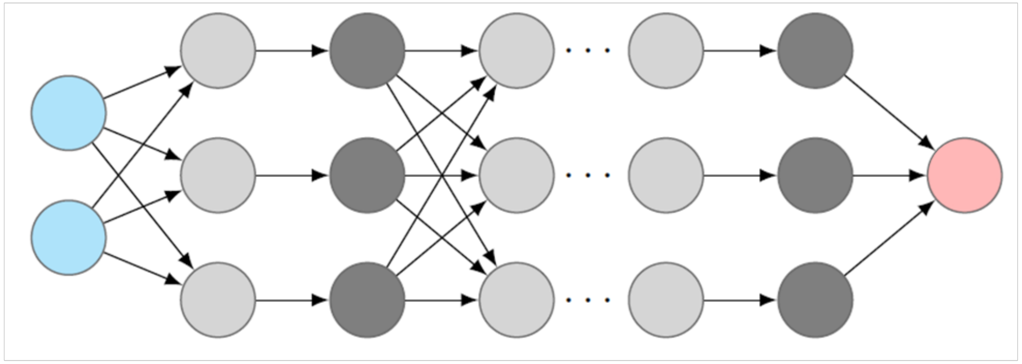
Convolutional Neural Networks

- Transfer and weakly supervised learning -

Lecturer: Michael Möller – michael.moeller@uni-siegen.de
Exercises: Hartmut Bauermeister – hartmut.bauermeister@uni-siegen.de

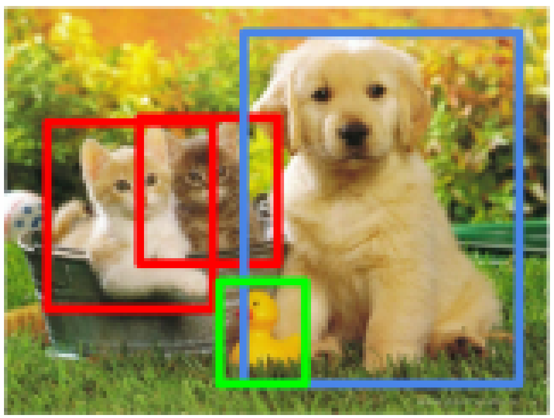


The problem of little data

Generating datasets for image classification is fairly easy.

But as the task becomes more difficult, annotations become tedious.

Object Detection



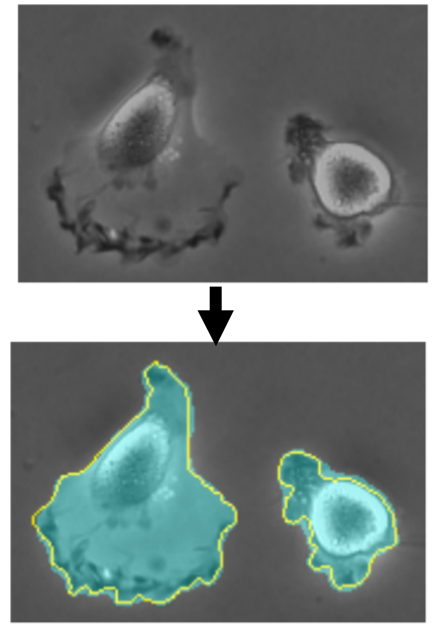
CAT, DOG, DUCK

Instance Segmentation

CAT, DOG, DUCK

https://leonardoaraujosantos.gitbooks.io/artificial-inteligence/content/object_localization_and_detection.html

Taken from "U-Net: Convolutional Networks for Biomedical Image Segmentation", 2015.



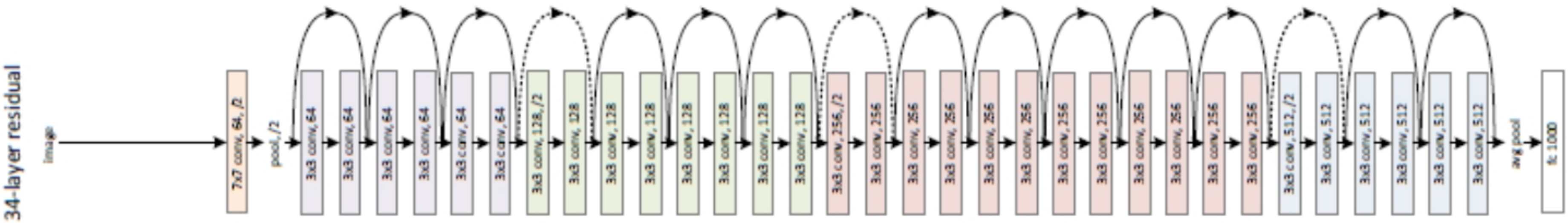
The situation is even worse if annotations require expert knowledge.

And even worse if it is difficult to acquire many exemplary images at all.

The problem of little data

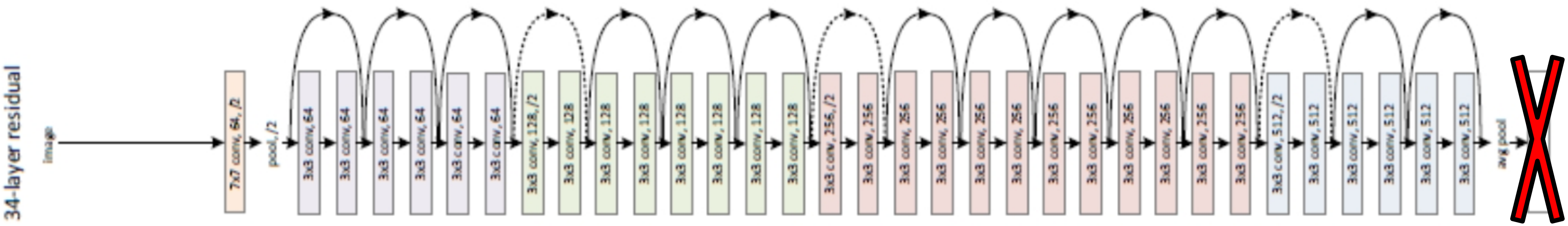
Girshik et al. “Rich feature hierarchies for accurate object detection and semantic segmentation”: “*In this paper, we propose a simple and scalable detection algorithm that improves mean average precision (mAP) by more than 30% relative to the previous best result on VOC 2012 […] when labeled training data is scarce, supervised pre-training for an auxiliary task, followed by domain-specific fine-tuning, yields a significant performance boost.*”

Idea: Start with a network that has been trained on a large image databased on a related task, e.g. classification.



Believe that the features one extracts from images (no matter which ones) via the convolutional layers are interesting and representative for understanding images in general.

The problem of little data



Remove the fully connected layer(s) and replace them by a layer that maps to a size of output you'd like your network to predict. Take all weights from the previous training and only initialize new layers randomly.

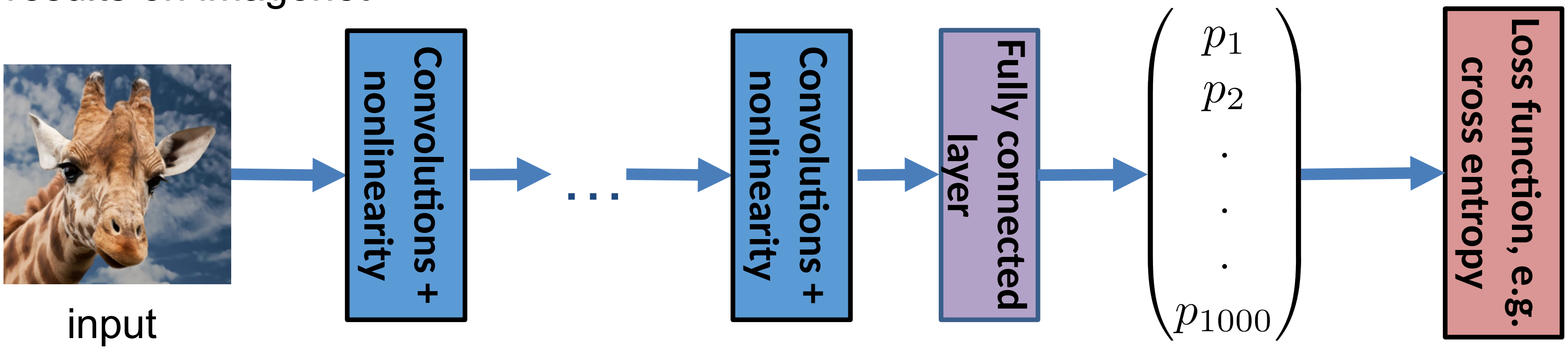
Train the network on your new task (with few training examples only), using

- a small learning rate on the pretrained layers, or
- freezing some or even all of the pretrained weights.

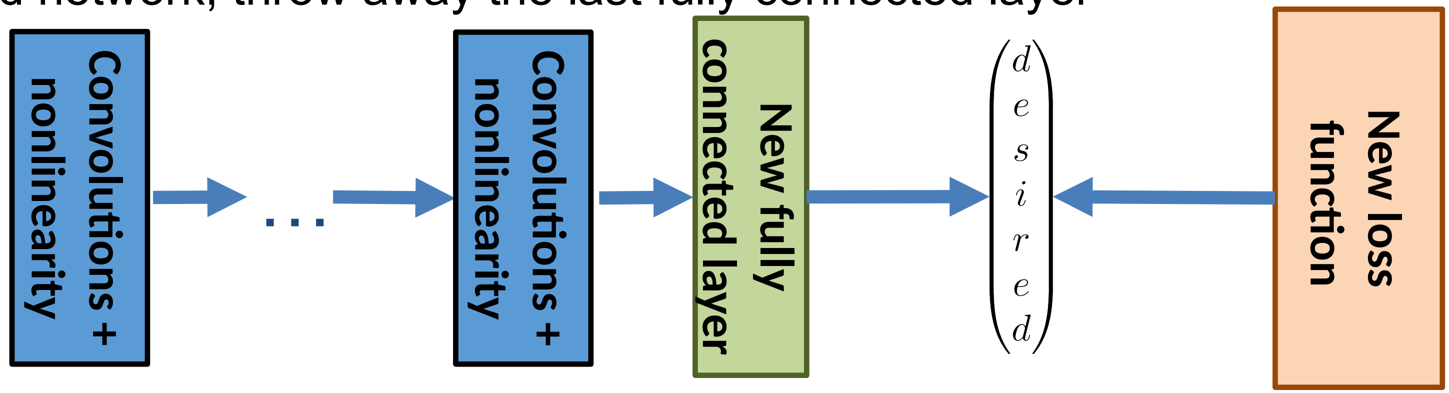
Researchers reported quite impressive performance boost and have utilized these ideas for a variety of different tasks. The general technique is known as ***transfer learning***.

In a nutshell

1. Train a network on a large annotated dataset, e.g. train for good classification results on imagenet



2. Copy the trained network, throw away the last fully connected layer



3. Assign a new layer, e.g. fully connected, and a new loss.

4. Train new network on little data, possibly freeze (or reduce stepsize for) convolutional weights.

Demonstration

Live demo with a webcam (in matlab)

Michael Möller – michael.moeller@uni-siegen.de

Transfer Learning

I have of course only given you a glimpse at the idea of transfer learning. Some more specific fine-tuning ideas and decisions how (and for which tasks) one pretrains, certainly help.

Some related techniques (for you to have heard the names) are

- *Domain adaptation*, where the task remains the same but the underlying data distribution changes. Example from Goodfellow, Bengio,Courville in “Deep Learning”: Sentiment prediction on a) reviews on books, and subsequently on b) comments on consumer electronics.
- *One-shot learning*, which is transfer learning with only one new example.
- *Zero-shot learning*, which is transfer learning without any new example. This of course requires some assumptions or prior knowledge.

Future of transfer learning?

Interesting preprint from Nov. 21st, 2018: He, Girshick, Dollar, "Rethinking ImageNet Pre-training".

"We report competitive results on object detection and instance segmentation on the COCO dataset using standard models trained from random initialization. The results are no worse than their ImageNet pre-training counterparts [...]."

Some general conclusions:

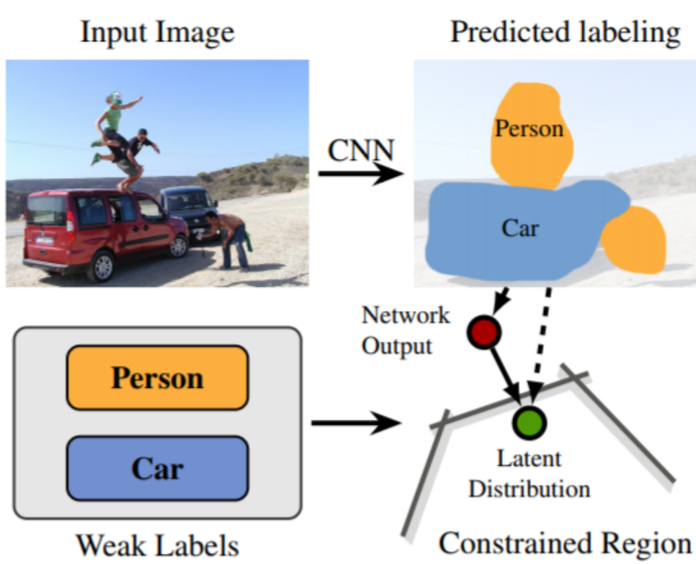
- Pre-training speeds up the convergence
- Pre-training does not improve the accuracy if sufficient (but still CV-small) training sets on the new task are available.
- Pre-training does improve the accuracy if little training data is available.
- Practically, sometimes spending more time on annotating data could be more useful than spending a lot of time on looking for datasets to use for transfer learning.

Dealing with little data

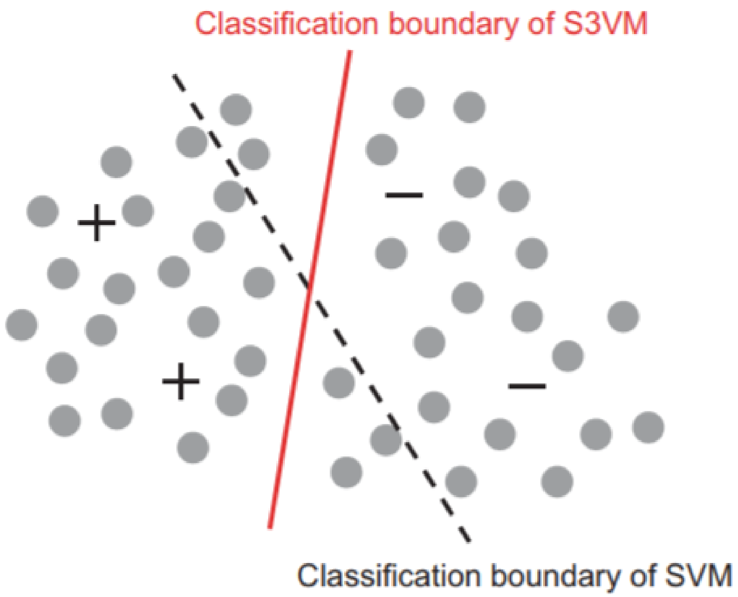
Weakly- or semi-supervised learning: Your data does not contain all the information (e.g. labels+additional assumptions or bounding boxes instead of segmentations).

Active Learning: The network gets little data but is allowed to query an oracle (human observer) for labels on certain (as few as possible) examples.

Partially annotated data: Only a few examples have labels.



From: Pathak et al., "Constrained Convolutional Neural Networks for Weakly Supervised Segmentation"



From: Zhou, "A brief introduction to weakly supervised learning"