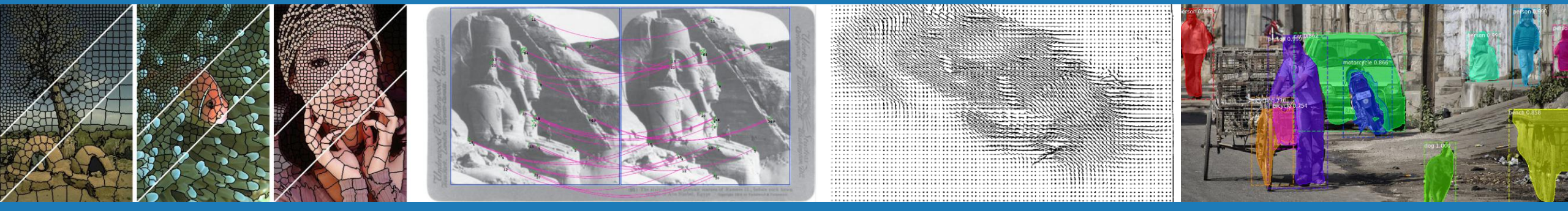
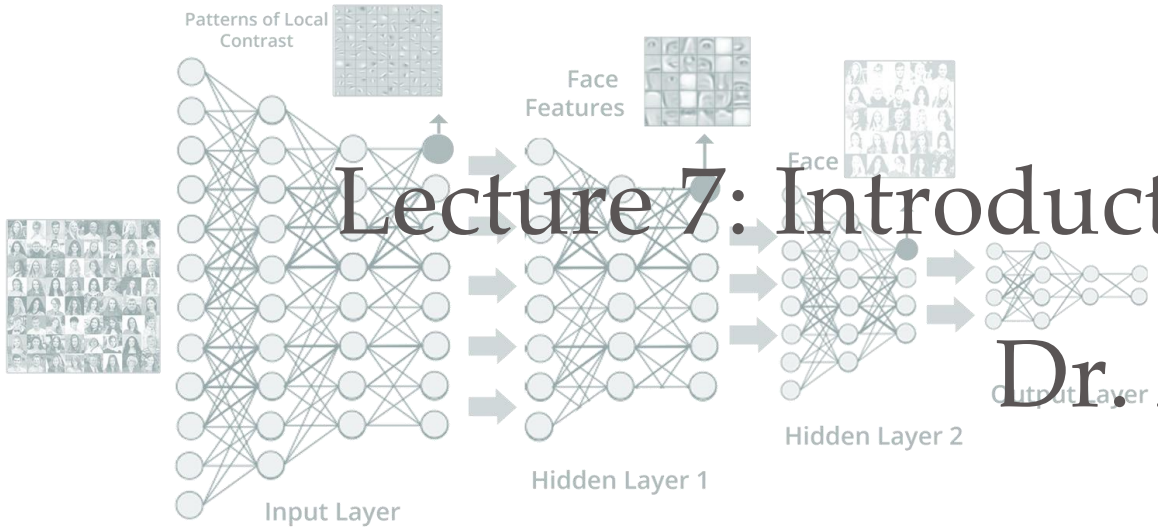
INSTITUTE FOR MACHINE LEARNING AND ANALYTICS



Computer Vision

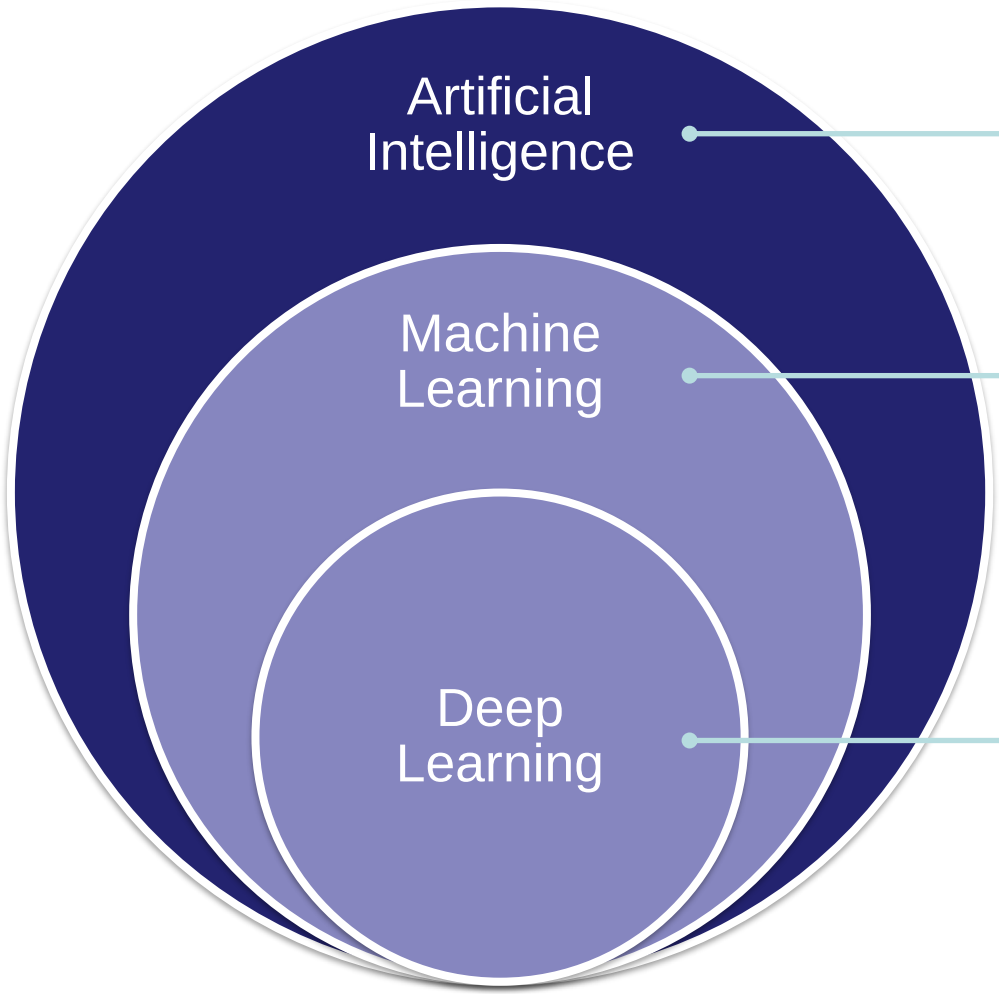


Lecture 7: Introduction to DL-based Methods

Dr. Xiao Zhao

EMI Elektrotechnik, Medizintechnik und Informatik

Artificial Intelligence



Artificial Intelligence (AI)

– Any techniques which make computers to mimic human beings

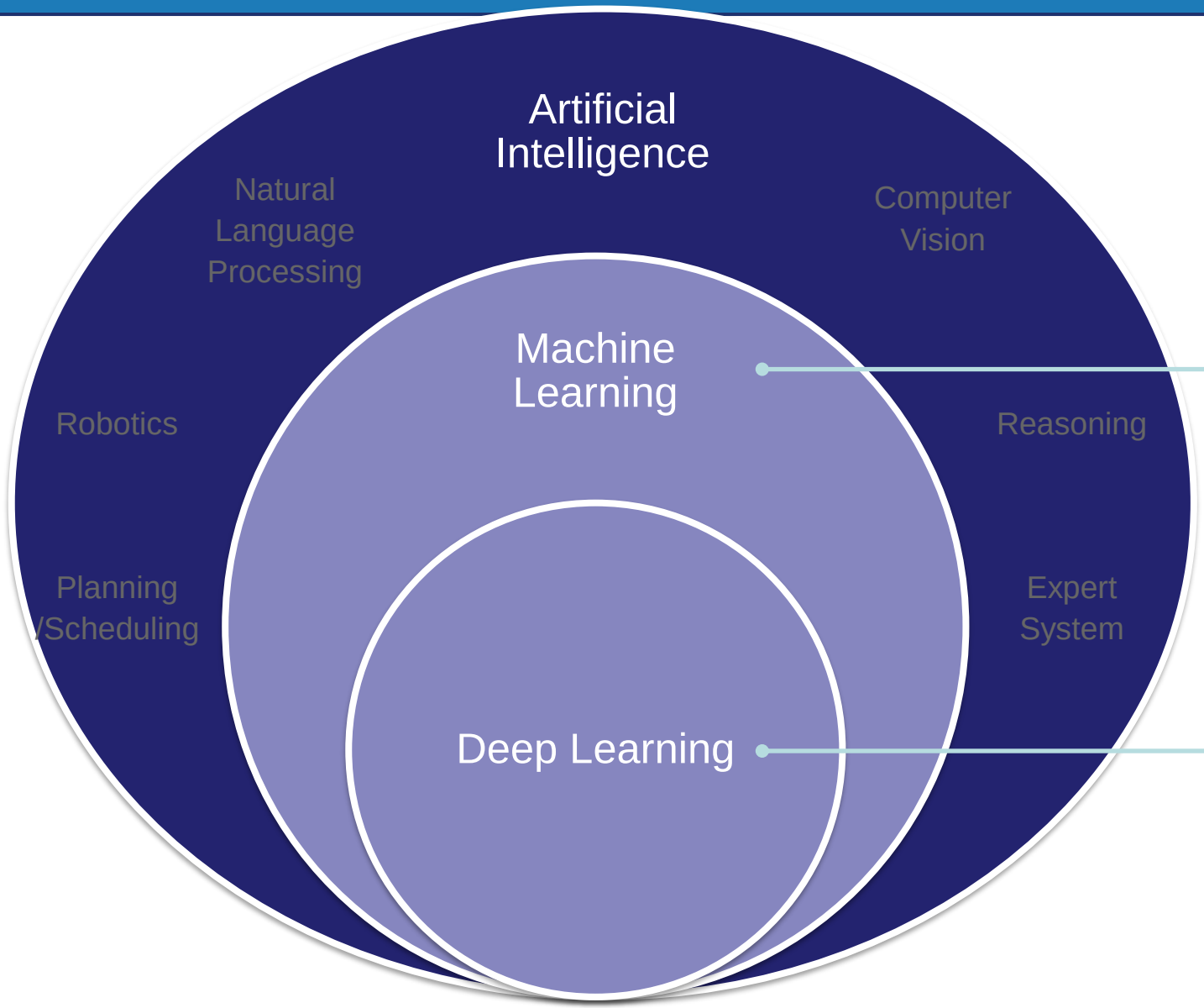
Machine Learning (ML)

– A subset of AI
– Techniques which use statistical methods to enable machines make human-like decisions

Deep Learning (DL)

– A subset of ML
– Techniques which use **multi-layer neural networks**

Artificial Intelligence



Models for Machine Learning

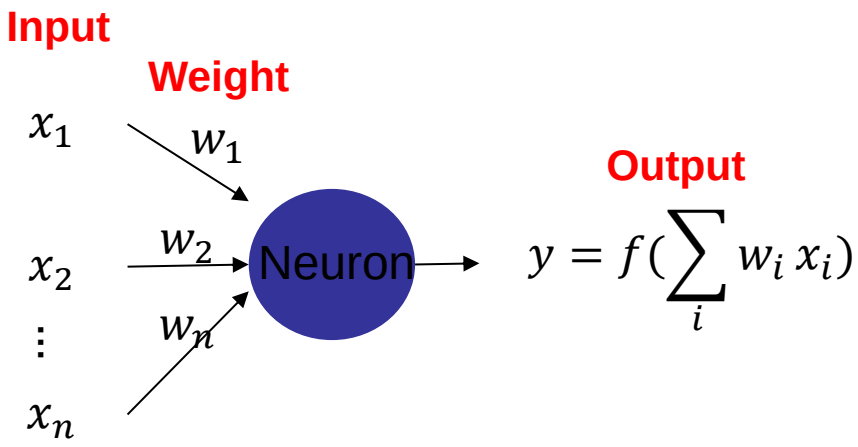
- Regression
- Support vector machine
- Bayesian
- Decision tree
- Decision rule
- Knowledge graph

Models for Deep Learning

- **Deep Neural Networks**
  - Feed forward NN
  - RNN, LSTM
  - convolutional NN (CNN)
  - attention, self-attention

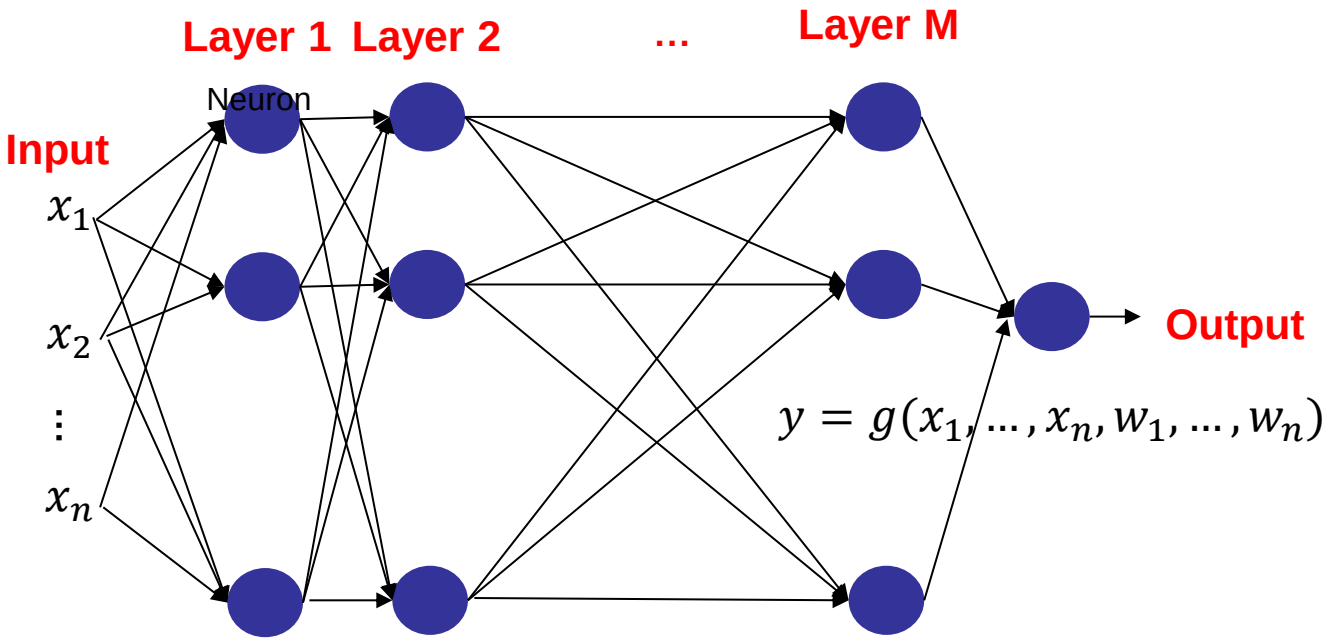
Neural Network

Single Neuron



- Neuron: multiple inputs, single output
- purely math. operations

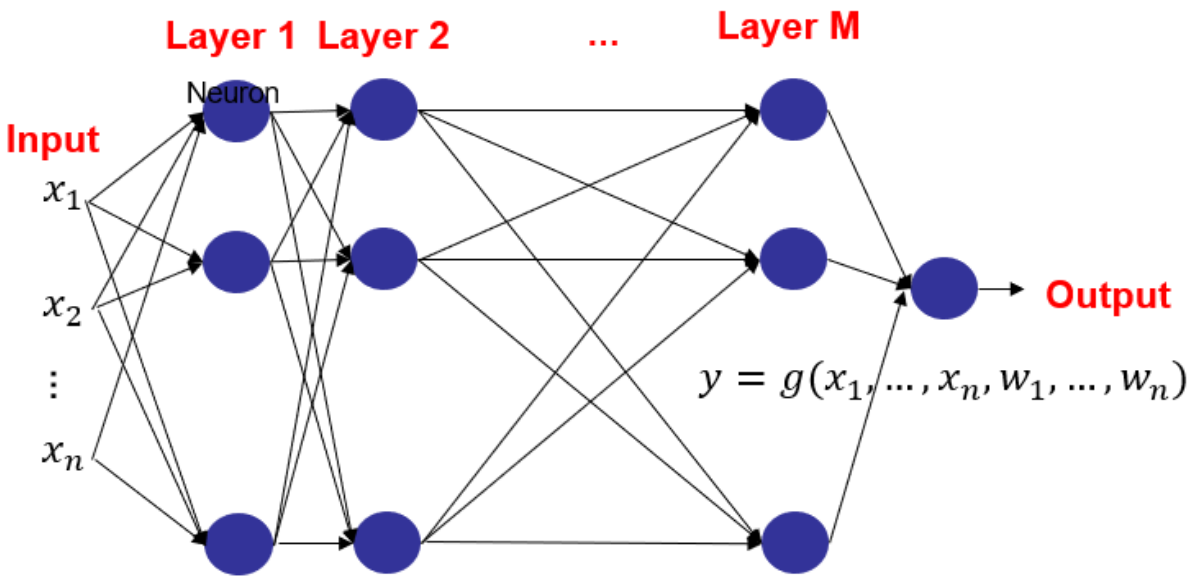
Neural Network



- Multiple neurons in multiple layers
- “Deep” Learning
- Still, purely math. operations

Training

- Neural Network



- Output y is a function of input x and network parameters w
- We aim to change w such that network's output y is the same the ground truth y^{GT}
- i.e. we wan to solve

$$\min_w L(x,w), \ where\ L(x,w) = |y(x,w) - y^{GT}|$$

Training = solving an optimization problem

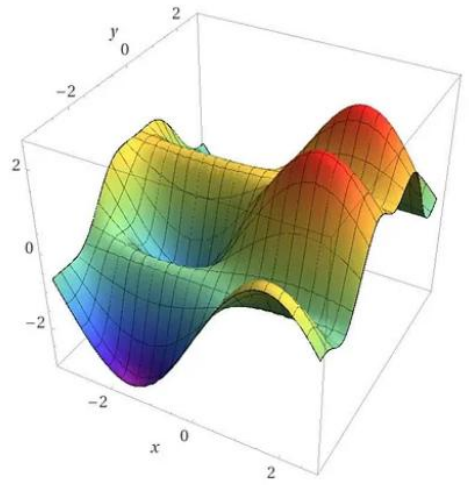
Training

Optimization Problems

- **Unconstrainted optimization**
  - min/max an objective function
  - No constraints appear

$$\min_{w} f(w)$$ **The Case of DL**

  - Example: $\min_{w_1,w_2} {w_1}^2 + w_2$



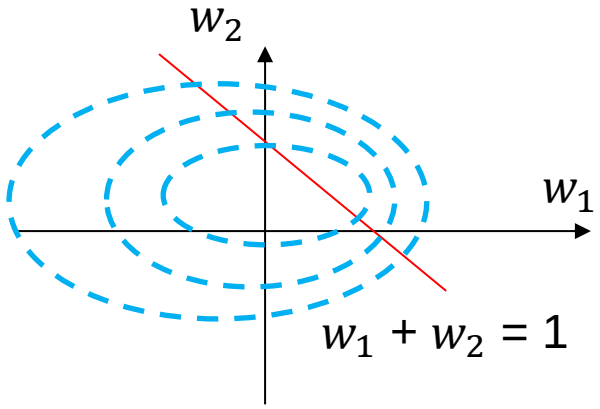
f(x, y)=−cos[(x−0.1)y]^2 −x sin (3x+y)

- **Constrained optimization**
  - min/max an objective function
  - Constraints appear

$$\min_{w} f(w)$$ objective fun.

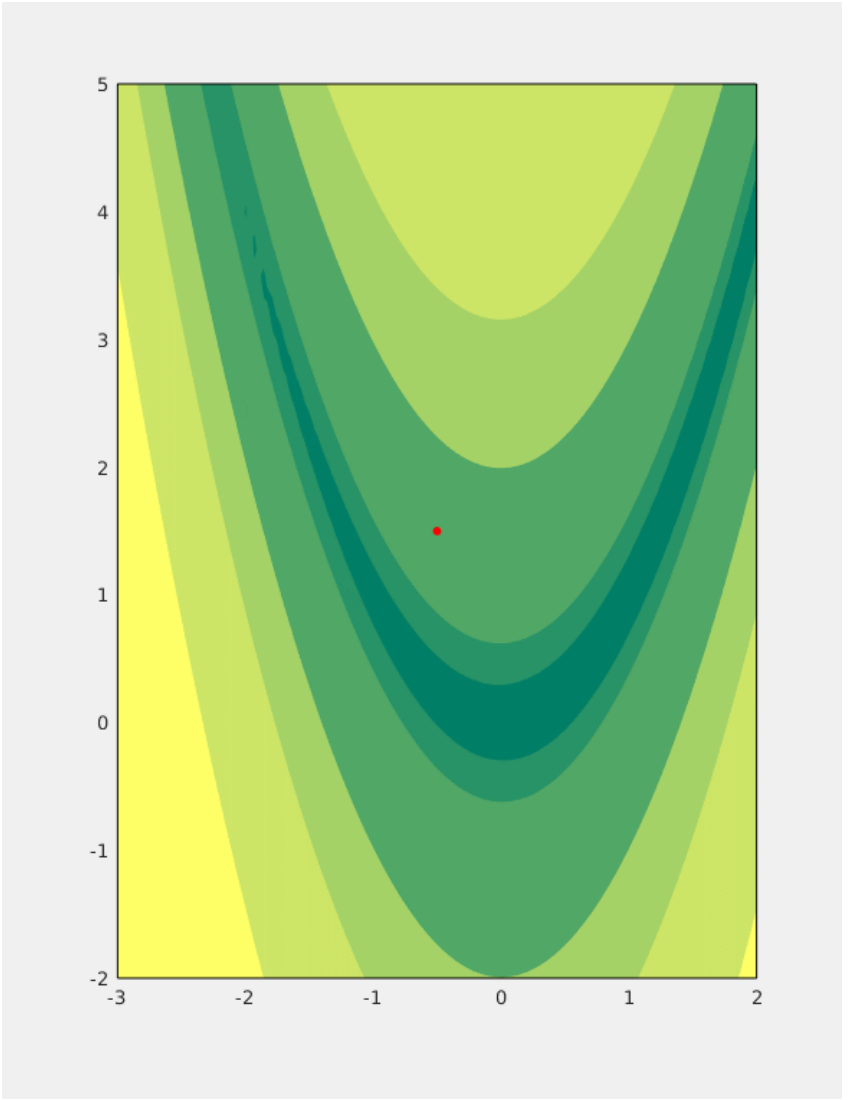
$$s.t.\ g(w) \geq 0$$
$$h(w) = 0$$ constrains

  - Example: $\min_{w_1,w_2} {w_1}^2 + w_2$
    s.t. $w_1 + w_2 = 1$



Algorithms for Optimization Problems

- **Derivative-free algorithms**
  - Do not use derivative information
  - Usage scenarios:
    - When derivatives are not defined, e.g. integer optimization
    - When derivatives are hard to compute
  - Usually based on random sampling of the variable space and then search

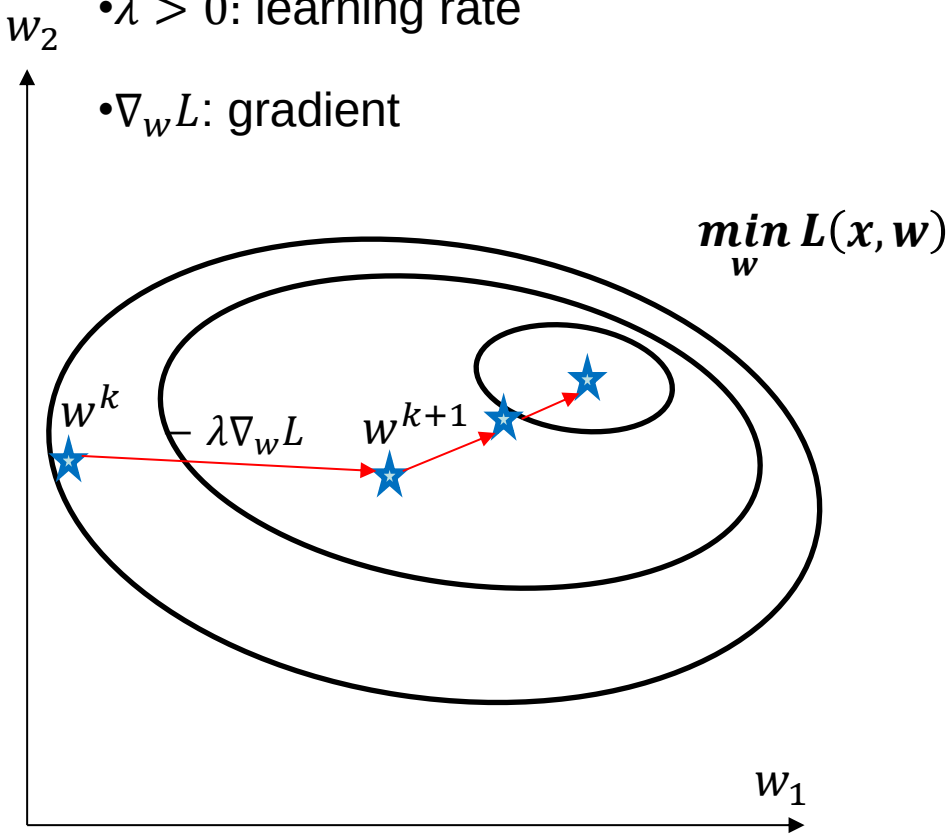


Source: https://en.wikipedia.org/wiki/MCS_algorithm

Algorithms for Optimization Problems

- **Derivative-based algorithms**
  - Based on derivative information
  - Variables are updated by derivative information, e.g.:
    - First-order gradients
    - Second-order gradients
    - Hessian matrix
  - Examples:
    - Gradient decent algorithm
    - Newton algorithm
    - Quasi-Newton algorithm

- **Gradient decent algorithm (no contraints)** :
  - $w^{k+1} = w^k - \lambda \nabla_w L(x, w)$
  - $\lambda > 0$: learning rate
  - $\nabla_w L$: gradient



Gradient Decent Algorithm

- How to obtain gradient $\nabla_w L(x, w)$?
- Method 1: Symbolic differentiation (**SD**)
- Method 2: Finite difference (**FD**)
- Method 3: Auto differentiation (**AD**)
  - Backpropagation is a special type of AD

- SD

$$f(t) = e^{t/2} \sin^2\left(\frac{t}{3}\right)$$

$$f'(t) = \frac{1}{2} e^{(t/2)} \sin^2\left(\frac{t}{3}\right) + \frac{2}{3} e^{(t/2)} \sin\left(\frac{t}{3}\right) \cos\left(\frac{t}{3}\right)$$

- FD:

$$f'(t) = \frac{f(t+\Delta t) - f(t)}{\Delta t}$$

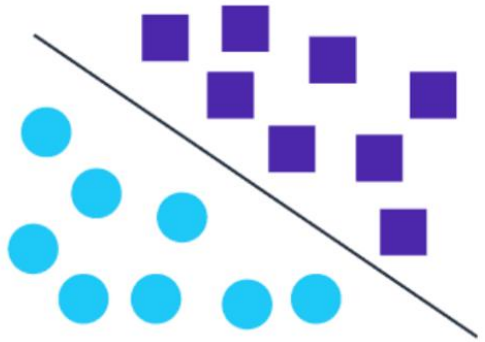
- AD:

$$y = f(g(h(x))) = f(g(h(w_0))) = f(g(w_1)) = f(w_2) = w_3$$
$$w_0 = x$$
$$w_1 = h(w_0)$$
$$w_2 = g(w_1)$$
$$w_3 = f(w_2) = y$$

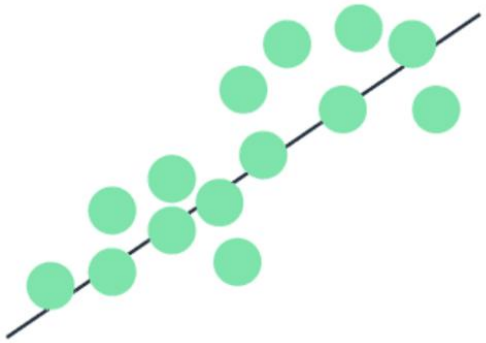
$$\frac{\partial y}{\partial x} = \frac{\partial y}{\partial w_2} \frac{\partial w_2}{\partial w_1} \frac{\partial w_1}{\partial x} = \frac{\partial f(w_2)}{\partial w_2} \frac{\partial g(w_1)}{\partial w_1} \frac{\partial h(w_0)}{\partial x}$$

Types of Training

- **Supervised Learning**
  - Labels y^{GT} are needed
  - Regression or classification task
  - Algorithms: decision trees, logistic regression, SVM



Classification Task,
e.g. object classification

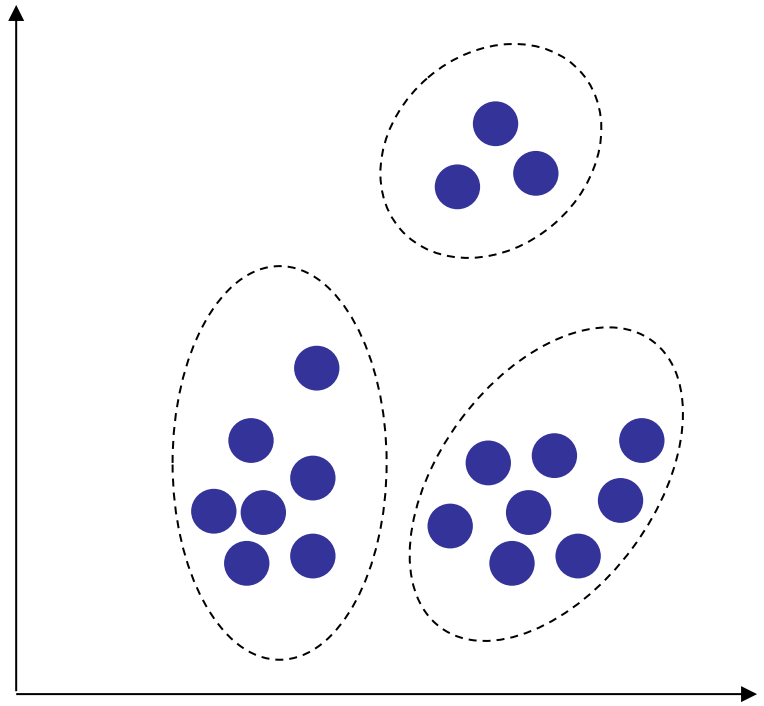


Regression Task
e.g. object detection
(Bounding box estimation)

Types of Training

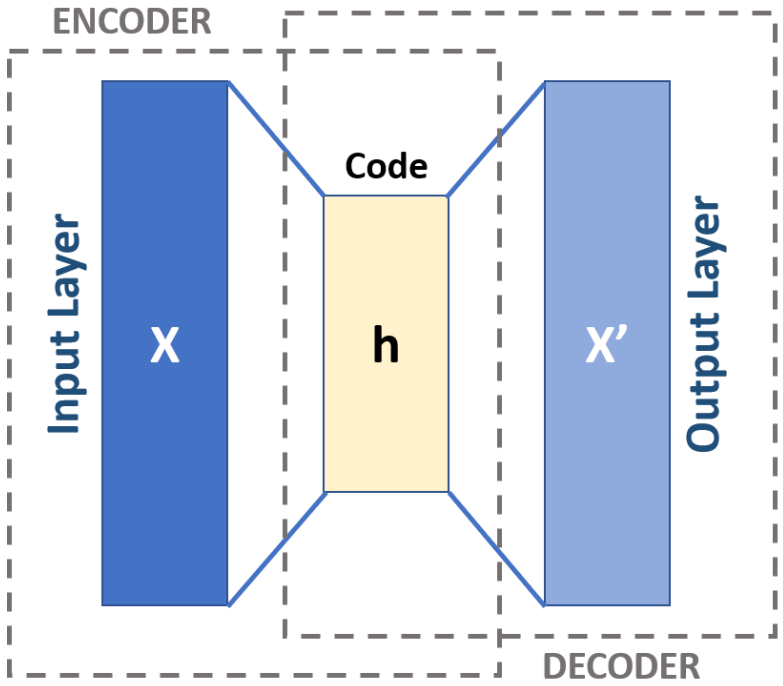
- **Unsupervised Learning**
  - Label y^{GT} do not need / exist
  - e.g. clustering task, association learning
  - Algorithms: k-means clustering, hierarchical clustering, apriori algorithm

Clustering



Types of Training

- **Self-supervised Learning**
  - Label y^{GT} do not exist explicitly
  - y^{GT} can be created using input x
  - e.g., y=x'=f(x), y^{GT}=x
  - Min loss(x', x)
    - When labels y expensive/hard to get
    - Pre-training a network with unlabeled data, e.g. GPT, BERT, MAE
    - Representation learning, e.g. autoencoder
    - Others?



e.g. Autoencoder

Self-supervised Learning

Advantage:

- No need to label the data
- Learn good initial parameters and pattern representations direct from data
  →meta learning

Usage Scenario:

- When labels y are expensive/hard to get
- Often used in the pretraining step of DNN, including:
  - Generative pre-trained transformers (GPT)
  - Vision Transformer (ViT)
  - Masked Autoencoder (MAE)

[1] OpenAI, GPT-4 Technical Report, ArXiv, 2024
[2] Dosovitskiy, et. al., An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, arXiv, 2021
[3] He, et. al., Masked Autoencoders Are Scalable Vision Learners, aXiv, 2021

Deep Learning Basic: Image classfication

- **Image classfication**: A core task in CV
- For each input image, decide its class from a pre-defined class set

→ **Cat**

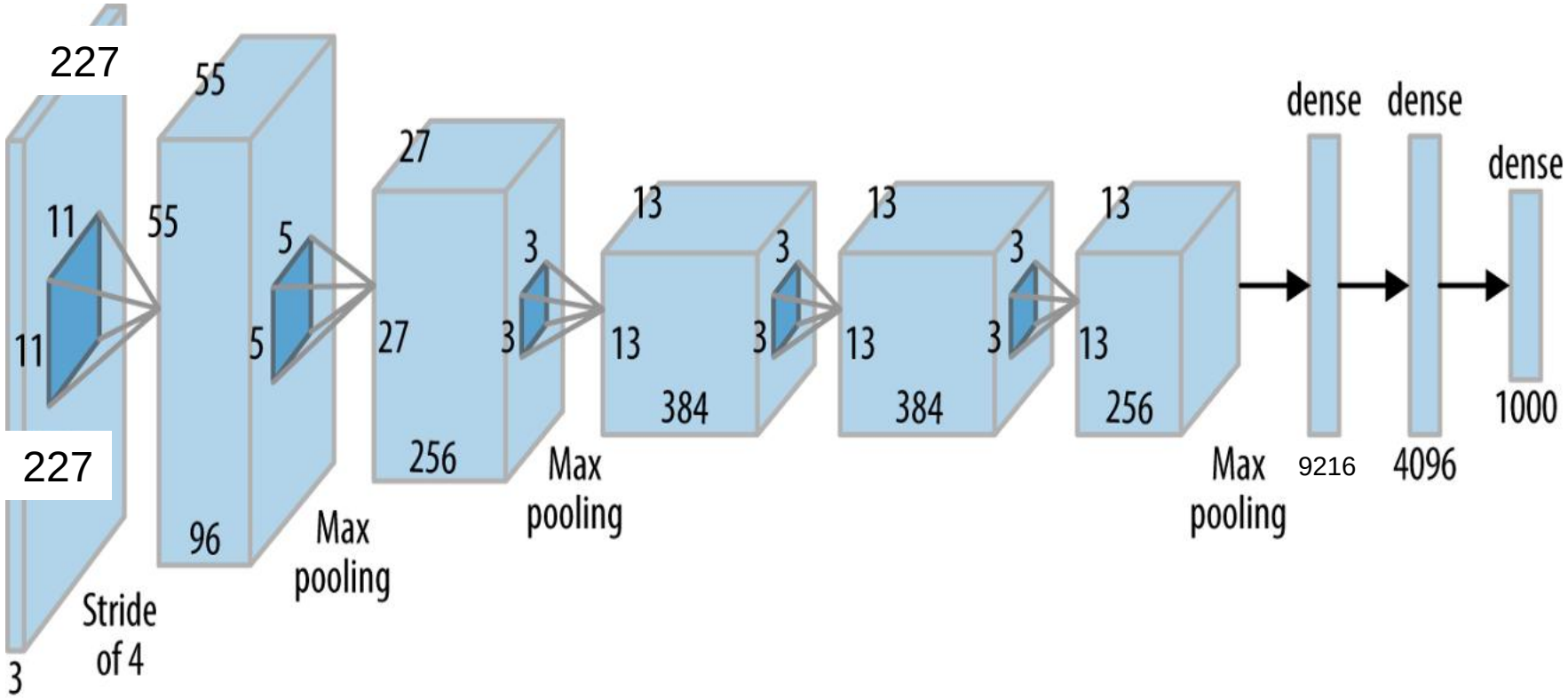
Source: https://www.image-net.org/

- No spatial information
- Only classes in the predefined class set can be predicted
  - COCO has 80 classes
  - ImageNet has 1000 classes

COCO: https://cocodataset.org/
ImageNet: https://www.image-net.org/

Image Classification: AlexNet

- AlexNet



- Input: $W \times H \times 3$
- Output: $y \in \mathbb{R}^{1000}$

Image: https://towardsdatascience.com/the-w3h-of-alexnet-vggnet-resnet-and-inception-7baaaecccc96

[1] Krizhevsky, et al., ImageNet Classification with Deep Convolutional Neural Networks, NIPS, 2012

Image Classification: ResNet

- ResNet (Residual Network)

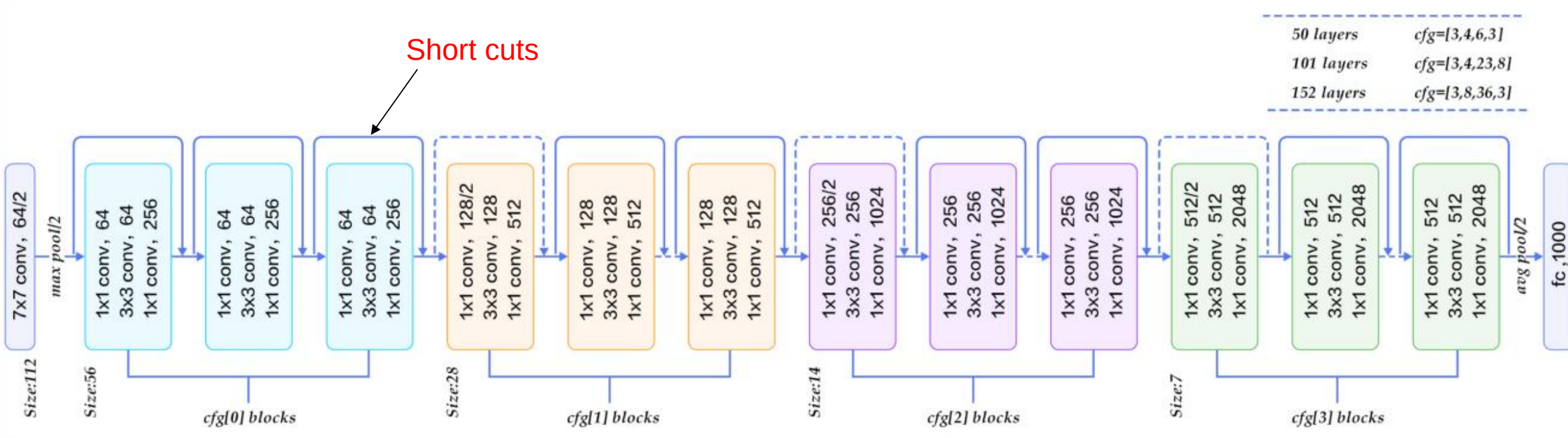
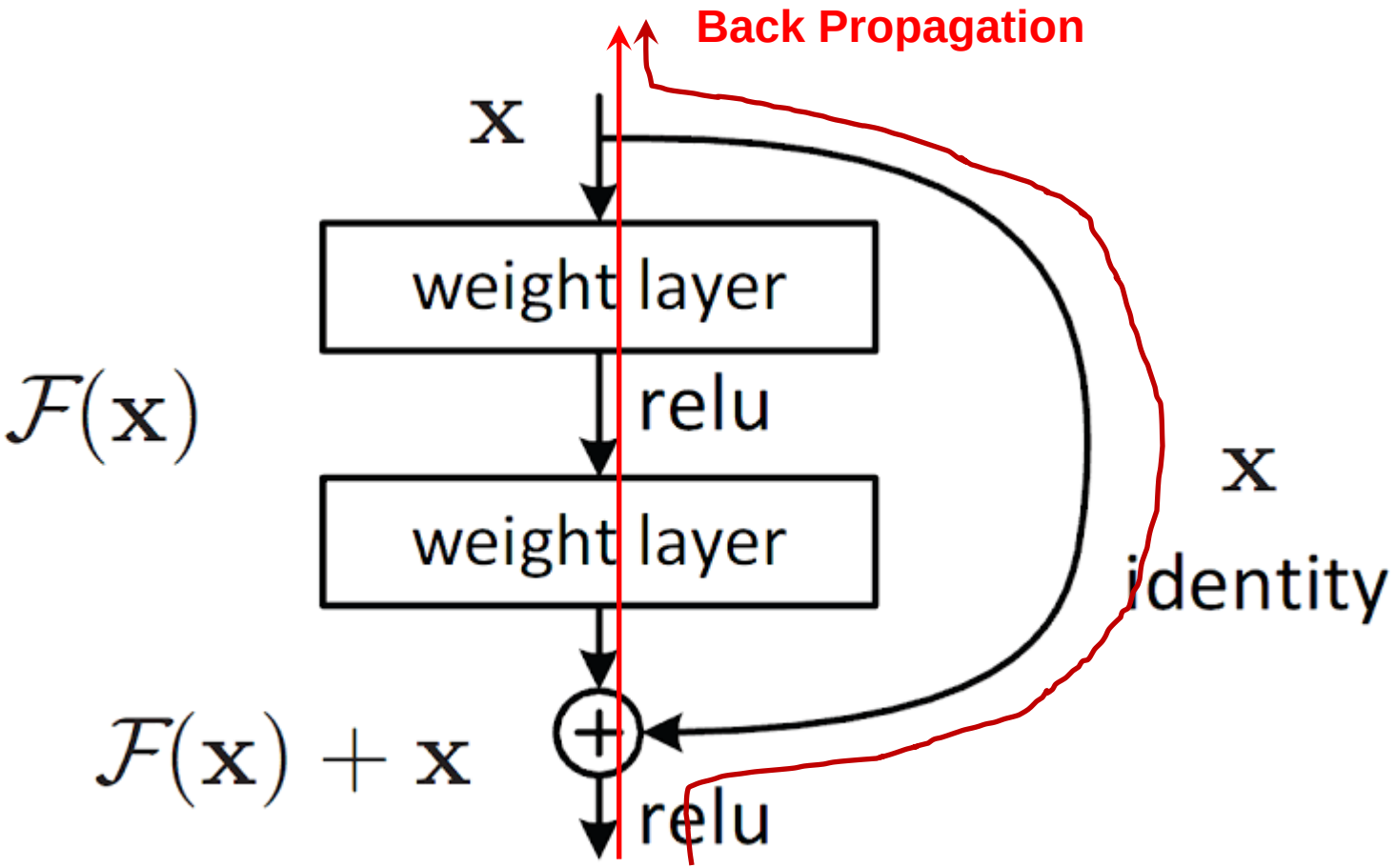


Image: https://miro.medium.com/max/2800/0*pkrso8DZa0m6IAcJ.png

[1] He, et al., Deep Residual Learning for Image Recognition, CVPR, 2015

ResNet: Residual Layer



$\mathcal{F}(\mathbf{x})$

$\mathcal{F}(\mathbf{x}) + \mathbf{x}$

Advantage:

- Shot cuts for gradient descent
- Trainability of lower-level layers

$$y = F(x) + x$$

$$\frac{\partial y}{\partial w_i} = F_x \frac{\partial x}{\partial w_i} + \frac{\partial x}{\partial w_i}$$

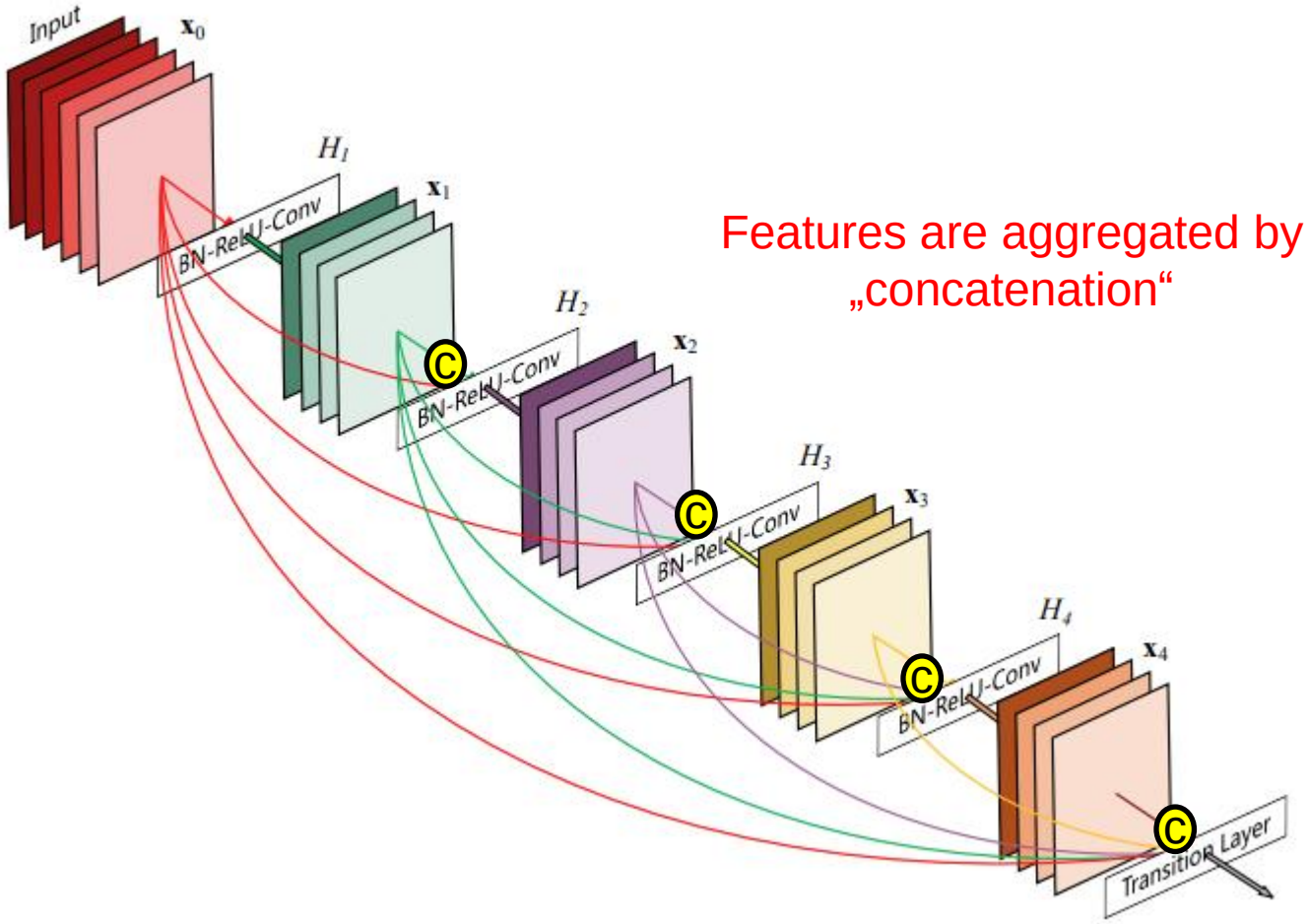
current layer lower-level layers

Features are aggregated by „summation“

ResNet

- By-pass routes offer a way to solve the problem of vanishing gradients.
- Gradients from lower levels can contribute to the total gradients along the network without going through the current layer.
- Even if the gradients of the current layer is zero, the total gradients along the full network is not necessarily zero

DenseNet



Features are aggregated by „concatenation“

A dense block with 5 layers

- Direct connections from any layer to all subsequent layers
- To further improve the information flow between layers

[1] Huang, et al., Densely connected convolutional networks, arXiv, 2018

DenseNet

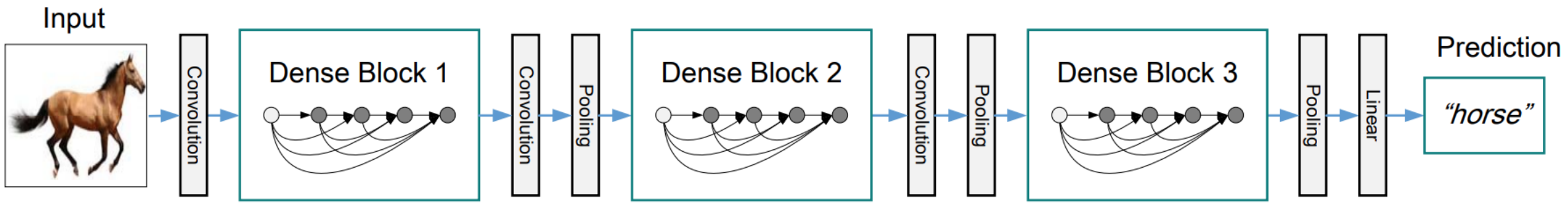
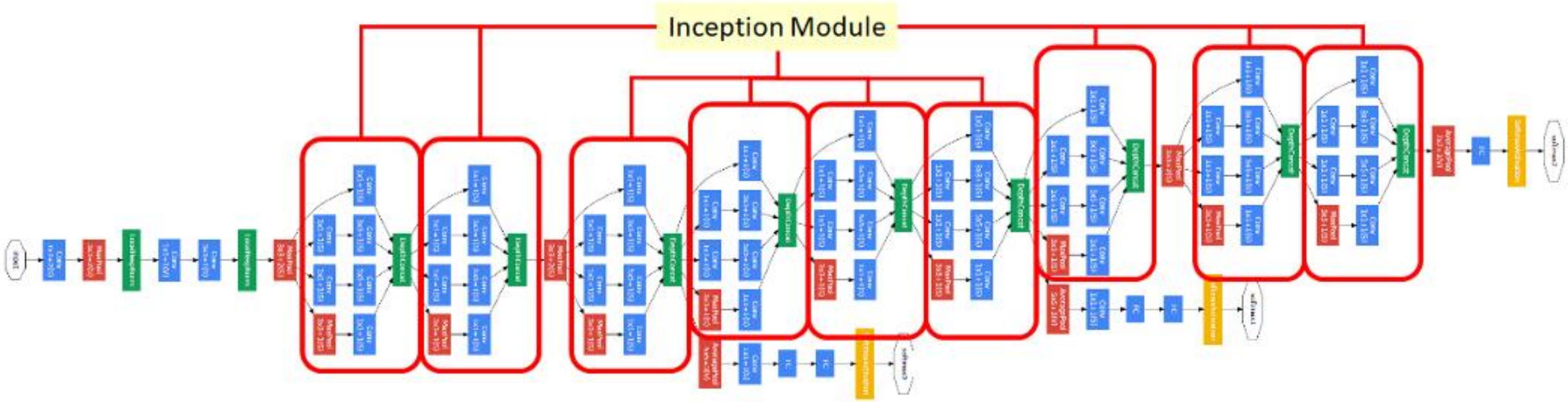


Figure 2: A deep DenseNet with three dense blocks. The layers between two adjacent blocks are referred to as transition layers and change feature-map sizes via convolution and pooling.

- Better performance than ResNet

[1] Huang, et al., Densely connected convolutional networks, arXiv, 2018

GoogLeNet / Inception network

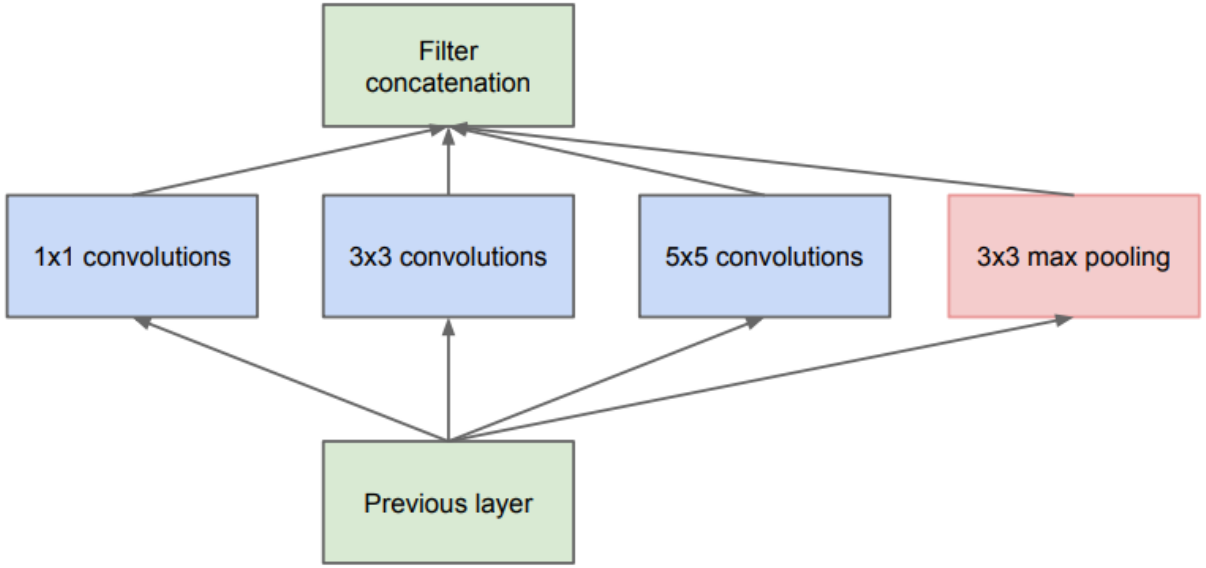


- GoogLeNet contains several inception modules
- Network gets deeper and wider

[1] Szegedy, et al., Going Deeper with Convolutions, CVPR, 2015

Inception Module

Inception Module



- Inception module contains kernels with different size
- Why useful?
  - Different field of view
  - “visual information should be processed at various scales” [1]

[1] Szegedy, et al., Going Deeper with Convolutions, CVPR, 2015

Field of View (FoV) of CNN

- Also called „ Receptive Field (RF)“
- FoV refers to the input region on a particular layer that affects a single pixel of the output layer.

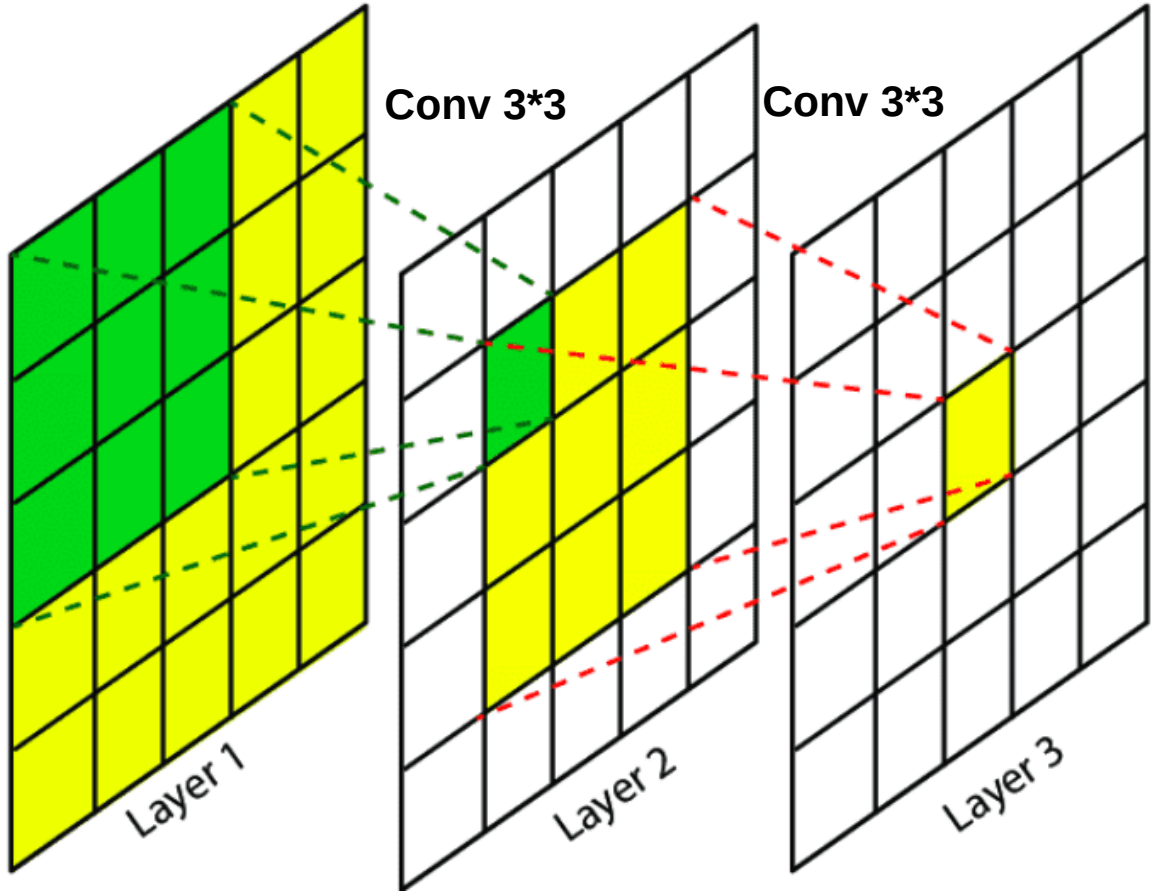


Image: https://theaisummer.com/receptive-field/

Objects in Different Scales

- The same object, but in different scales
- Can a network classify these images with the same B, H correctly?

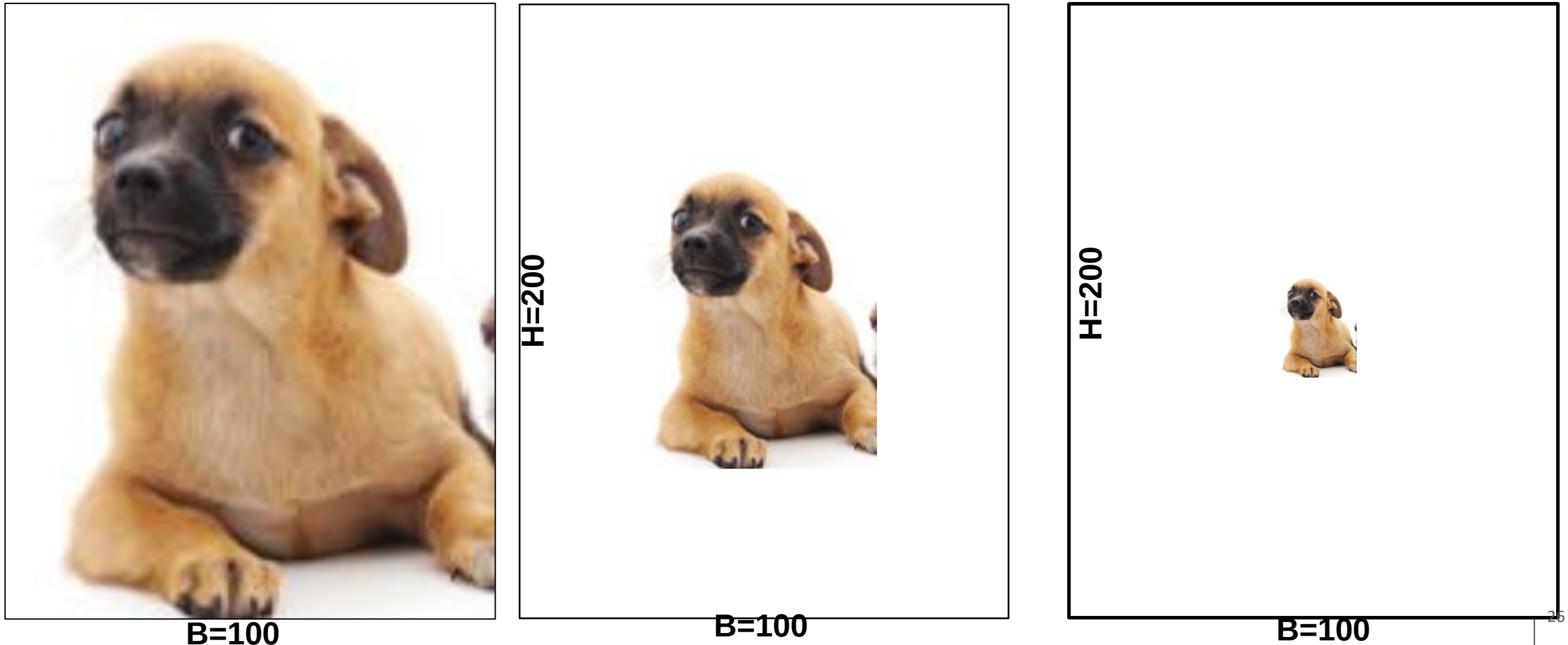
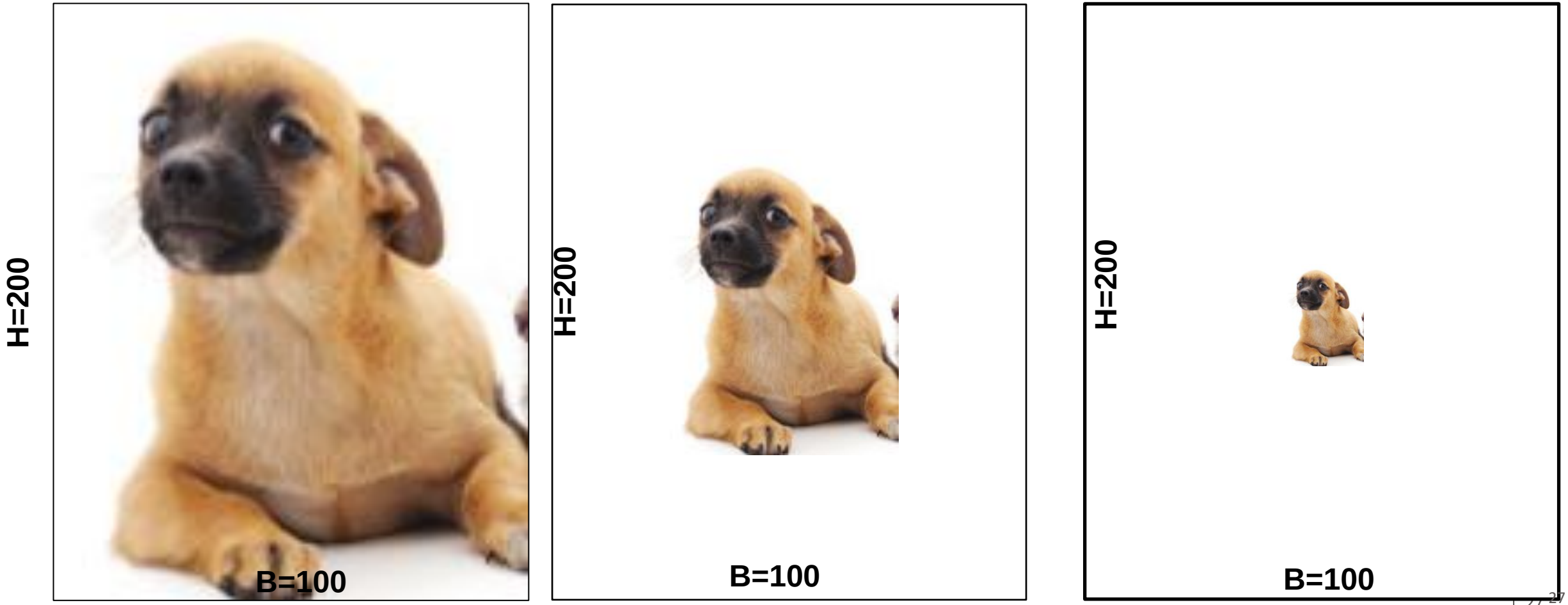
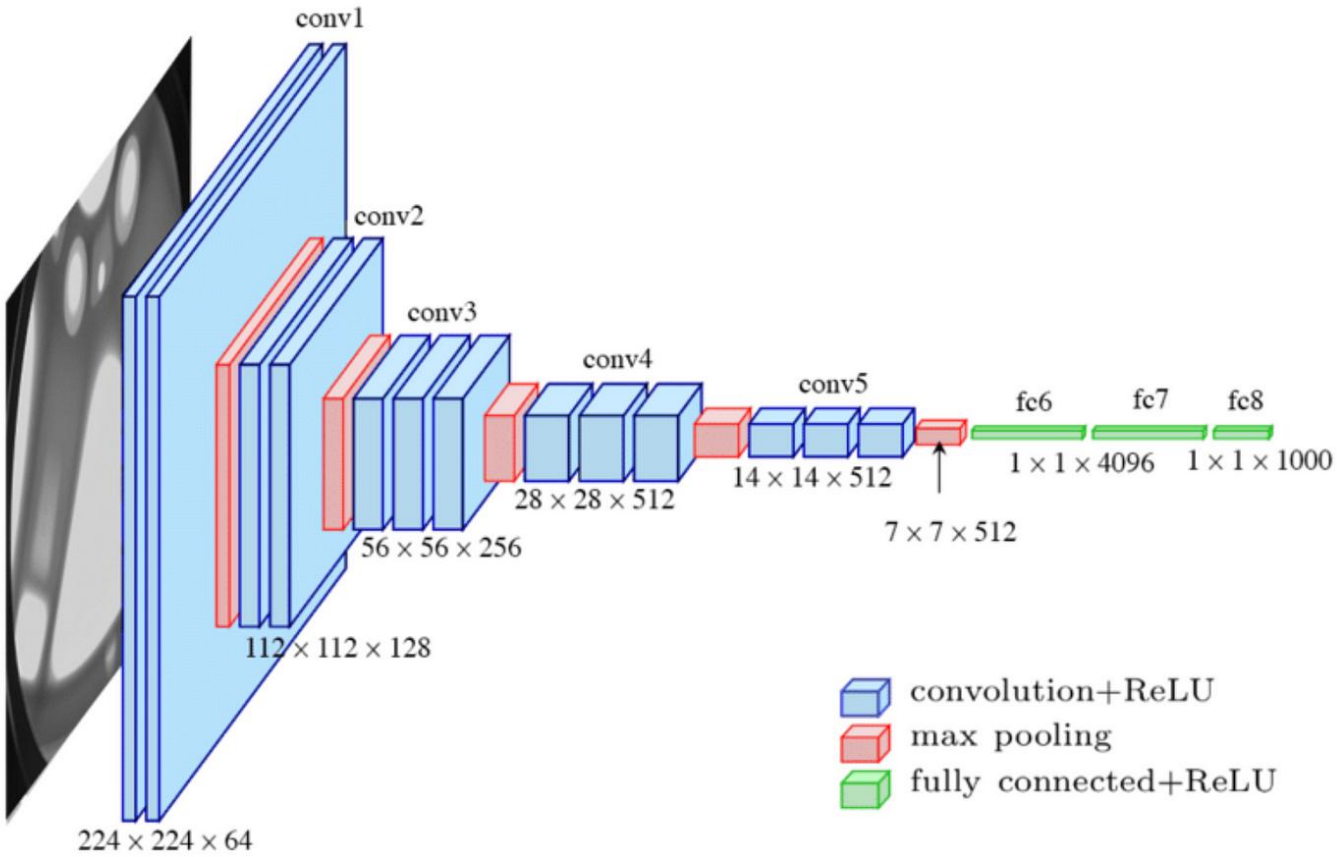


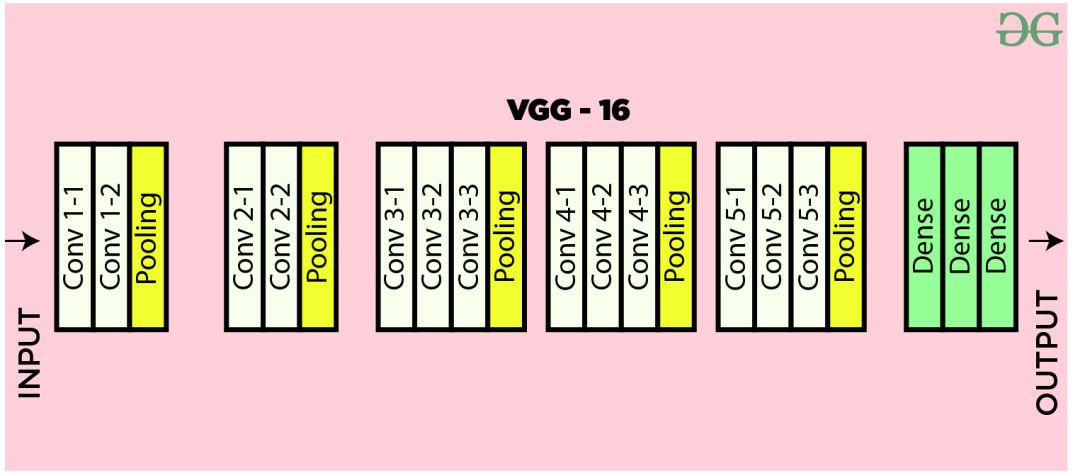
Image Classification

Larger FoV is needed



Labor: VGG-16





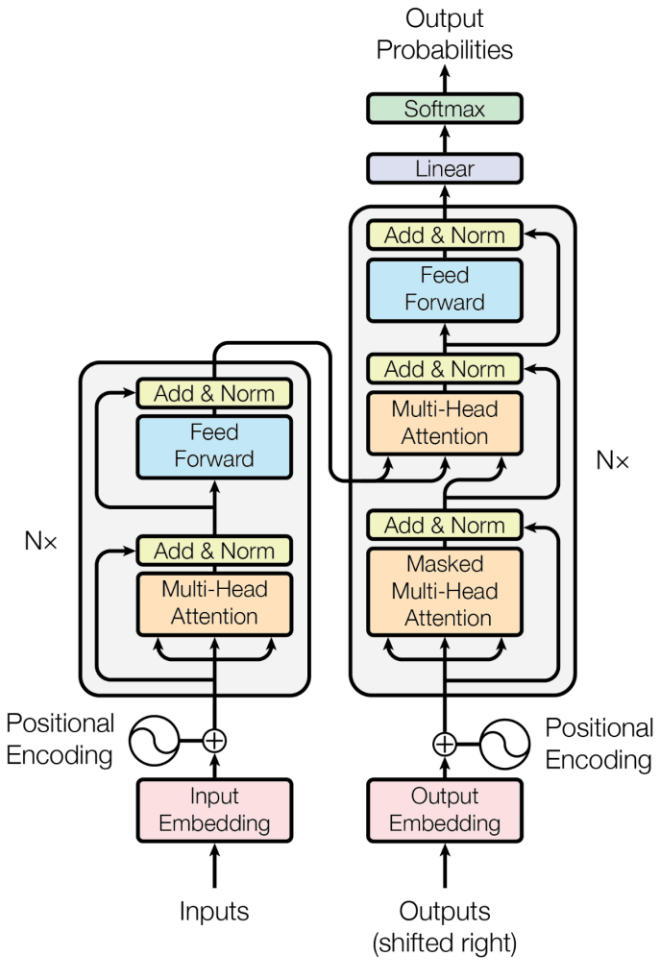
https://medium.com/@mygreatlearning/everything-you-need-to-know-about-vgg16-7315defb5918

Advanced Image Classification

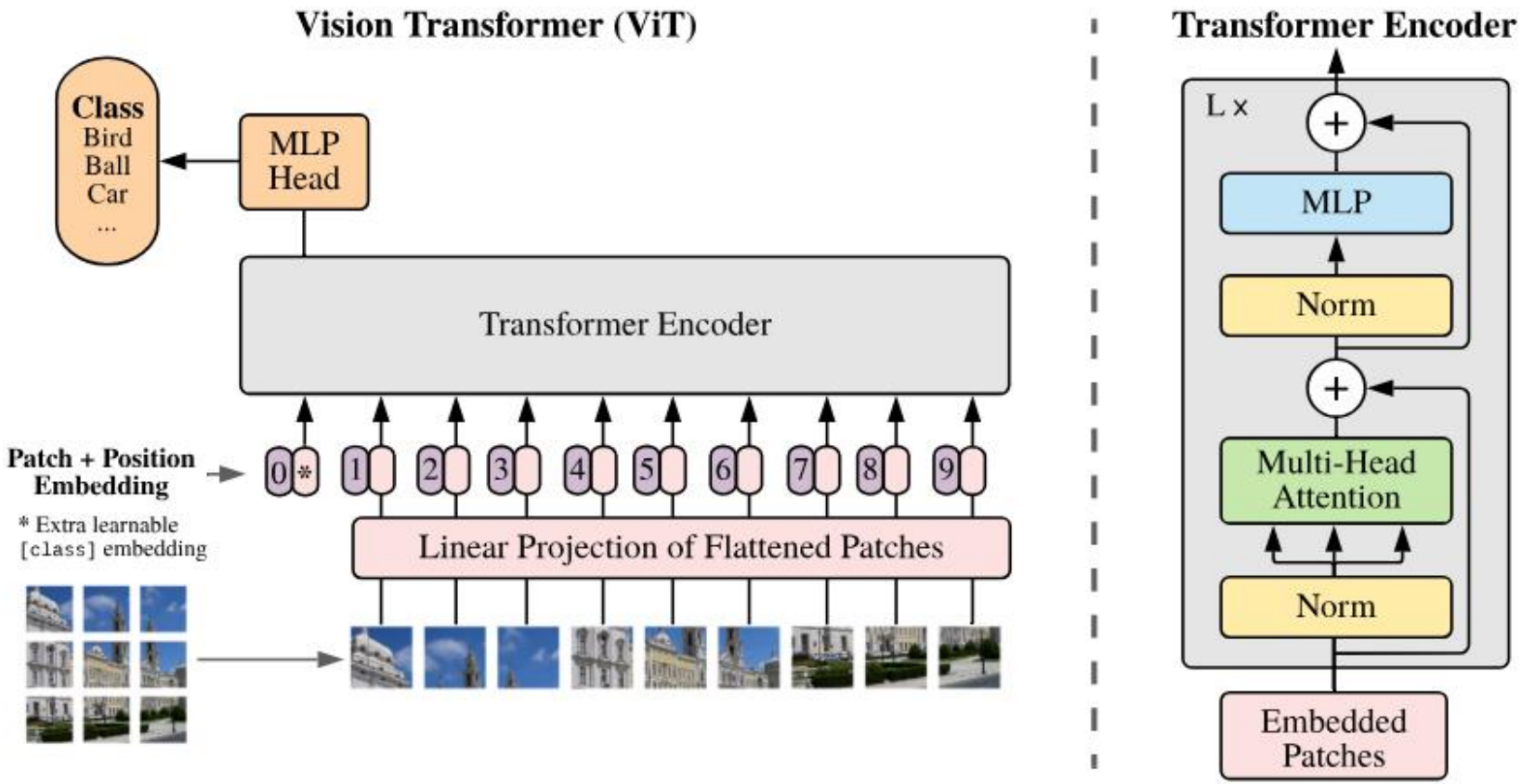
Transformer in NLP

- Transformer structure is based on attention mechanism
- Transformer contains an encoder and a decoder part
- an embedding layer transform input tokens to vectors/embedding (encoder part)
- Attention mechanism accepts a sequence of embeddings and outputs a sequence of embeddings **with the same length**
- Attention mechanism is a set operation (the sequence of embeddings does not play a role)
- Positional encoding is therefore needed to represent the sequence

Transformer, 2017



Vision Transformer (ViT)

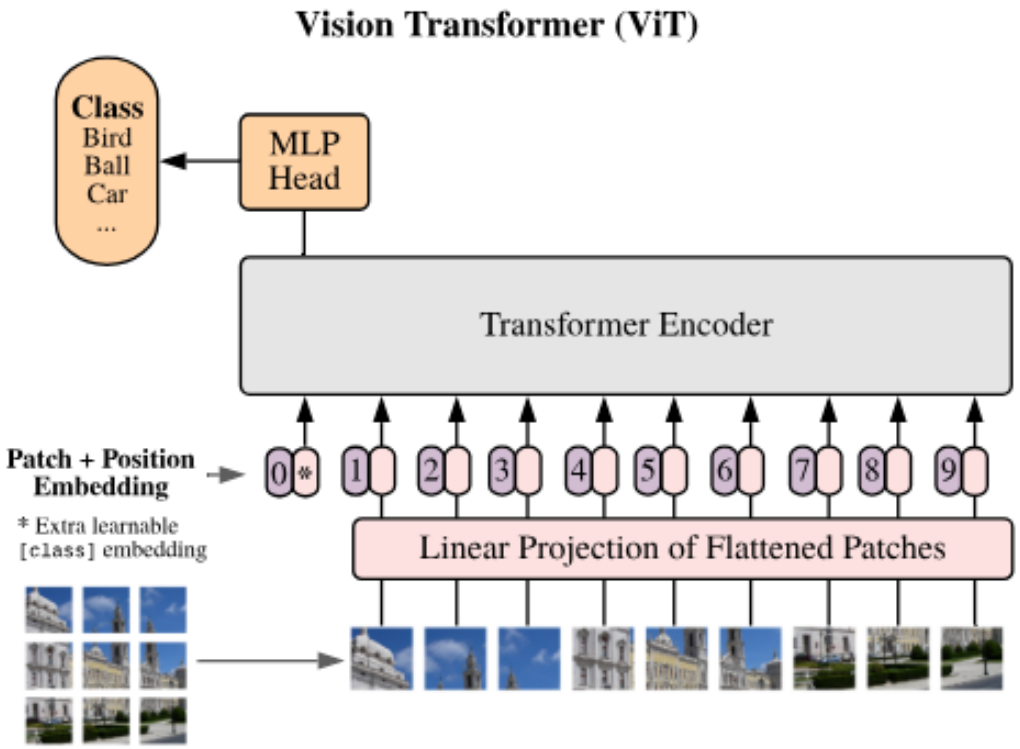


- Google‘s work in 2021
- Based on self-attension
- **No convolution**
- Used only the encoder of transformer

[1] Dosovitskiy, et. al., An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, arXiv, 2021

Main steps in ViT

- An image is divided into 16*16=256 patches
- Each 2D patch is flattened into a 1D vector
- Each 1D vector is projected by linear layers
- The derived embeddings are extended with position embeddings
- For each image, 256 vectors are derived. They are appended with a special token at „0“-position and then inputted to the transformer.
- The „0“-position of the output sequence is used for classifying.



ViT Summary

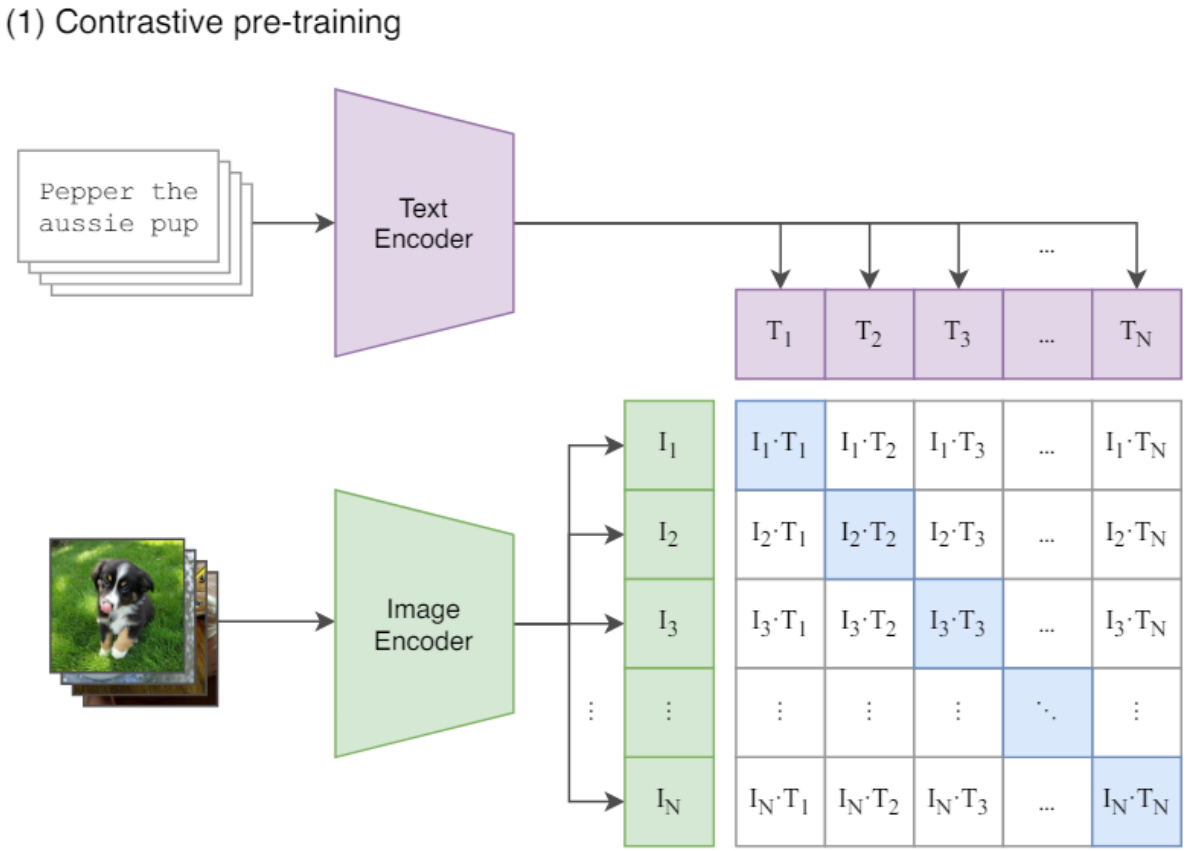
- Image has to be firstly patched, since transformer can not take long sequences.
- Performance of ViT is comparable to CNN, if training set are large.
- For small training set, better try CNN at first.
- Pretained ViT can be downloaded and then fine tuned for specific purposes.
- ViT suggests a possibility to treat images and languages in the uniform framework of 'attension'.
- Image and language are the same for computers!

Contrastive Language-Image Pre-Training (CLIP)

- OpenAI´s work in 2021
- Text-Image based **contrastive pretraining**
- **Zero-shot prediction** on new classes (next slide)

Advantage:

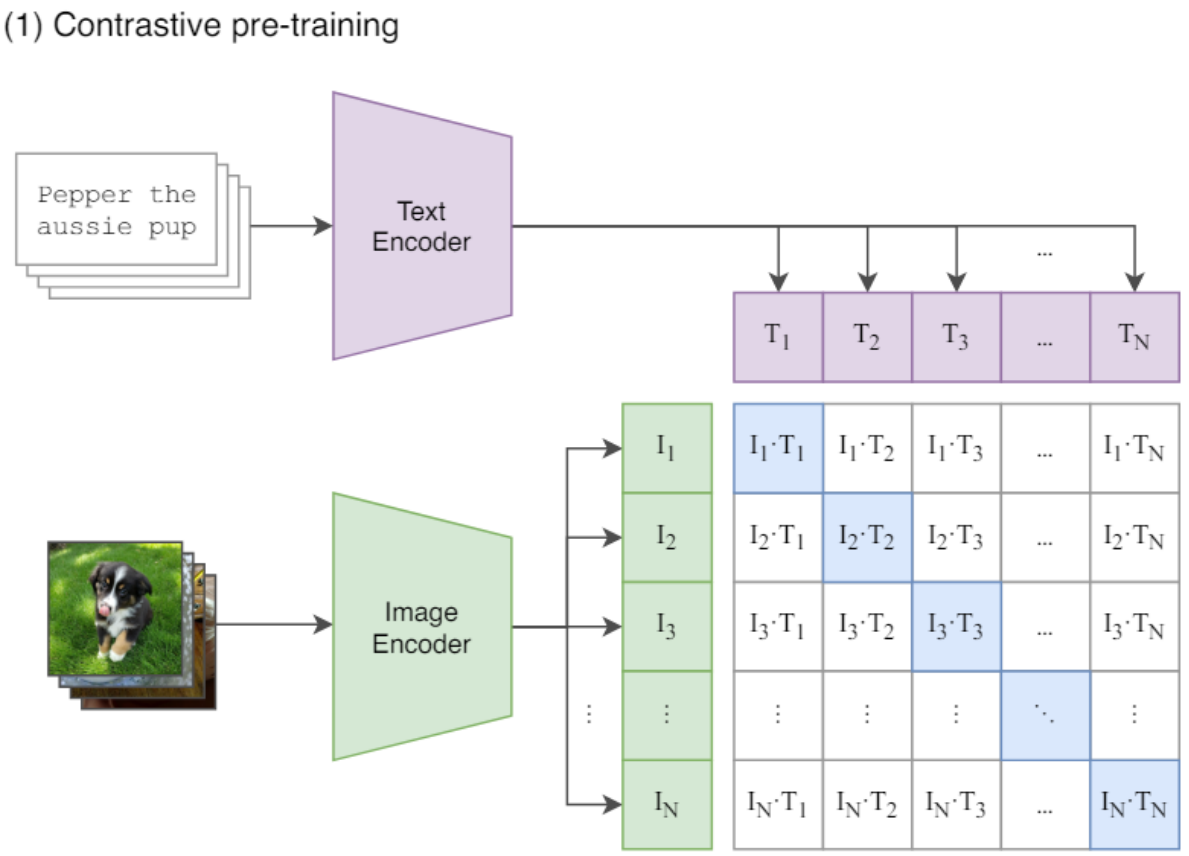
- CLIP can classify objects with any classes!
- Do not need to predefine classes during the training!



[1] Radford, et. al., Learning Transferable Visual Models From Natural Language Supervision, arXiv, 2021

CLIP: Contrastive Training

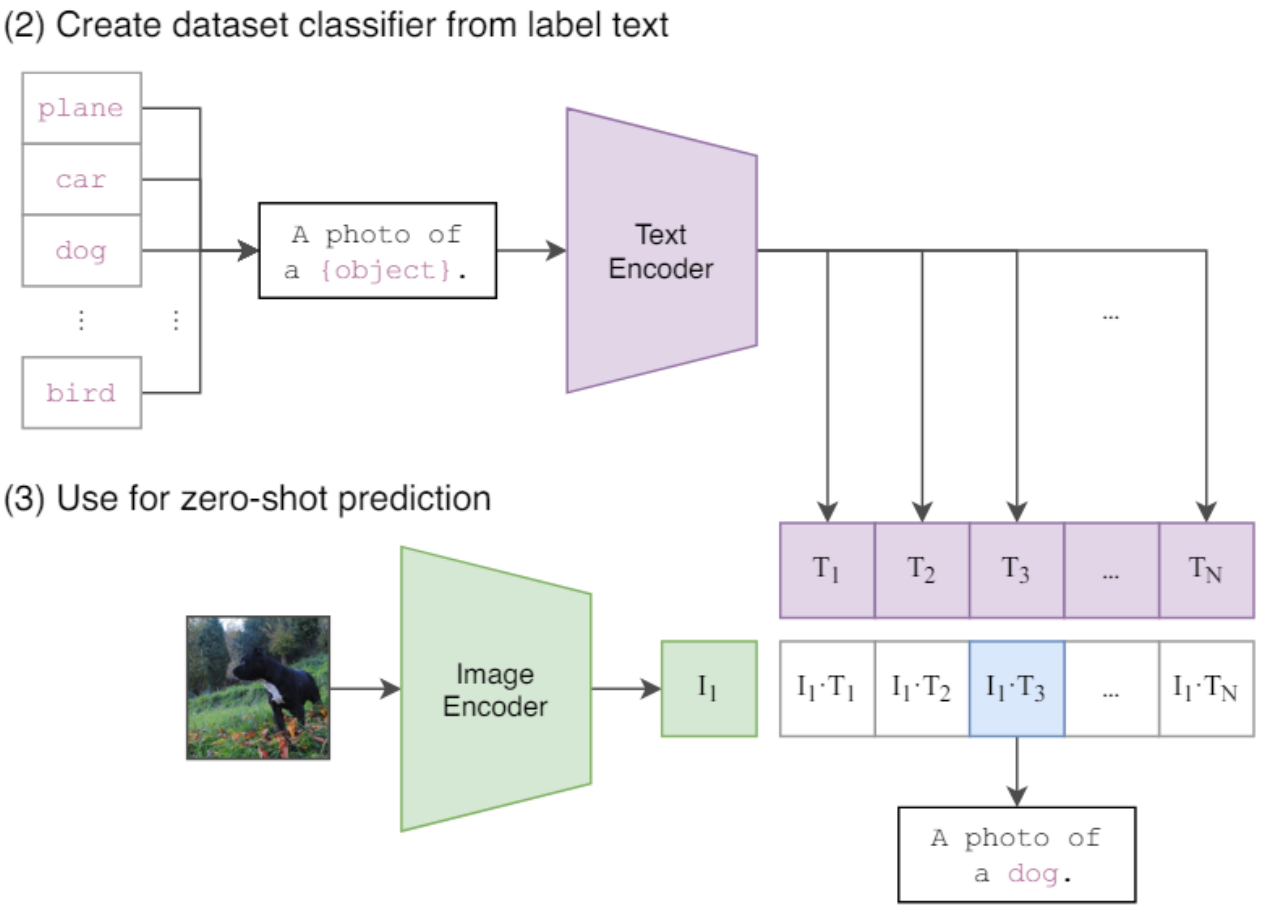
- Contrastive Training
  - Dataset: images with captions ("text")
  - A multimodal model:
    - Text encoder
    - Image encoder
  - The model learns the **similarity** between images and their captions
    - Assign high scores to ground-truth image-and-caption pair
    - Assign low value to other pairs
  - Representation learning



CLIP: Prediction

- **Zero-shot prediction:**
  - Prepare a list of candidate classes in text
  - Convert them into sentences, e.g. "*a photo of {object}*"
  - Compute the embedding of the image I_1 and the embedding of the sentence T_i
  - Compare similarity of embeddings by $I_1 \cdot T_i$
  - If $I_1 \cdot T_{i^*}$ is big enough, then class i^* is found

Zero-shot means that the model makes predictions during the test time on a data set, which come from a different domain than the training set.



CLIP

Food101

guacamole (90.1%) Ranked 1 out of 101 labels

✓ a photo of **guacamole**, a type of food.

× a photo of **ceviche**, a type of food.

× a photo of **edamame**, a type of food.

× a photo of **tuna tartare**, a type of food.

× a photo of **hummus**, a type of food.

https://openai.com/research/clip

CLIP

https://openai.com/research/clip

CLIP

- Task example:
- Given a photo of a smartphone. Describe the steps of using CLIP to classify the manufacturer of this handy? (e.g. apple, samsung, huawei, ...)
- If the performance is bad, which can be reasons? What can be done?

Tasks Beyond Image Classification

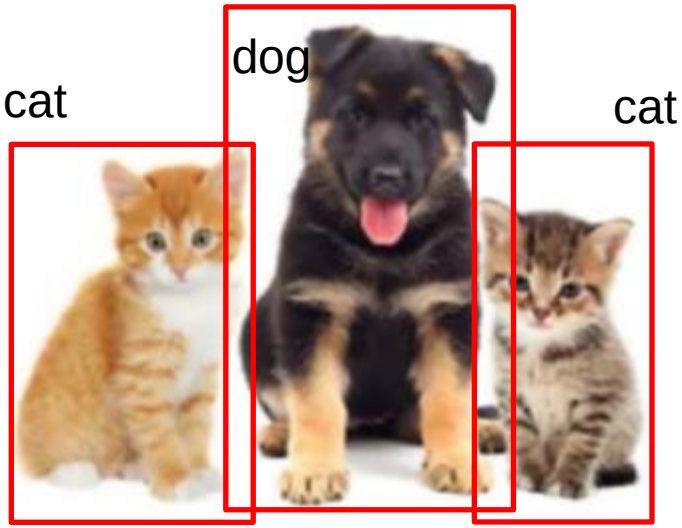
Tasks Beyond Image Classification

- **Object Detection**
- **Semantic Segmentation**
- **Instance Segmentation**

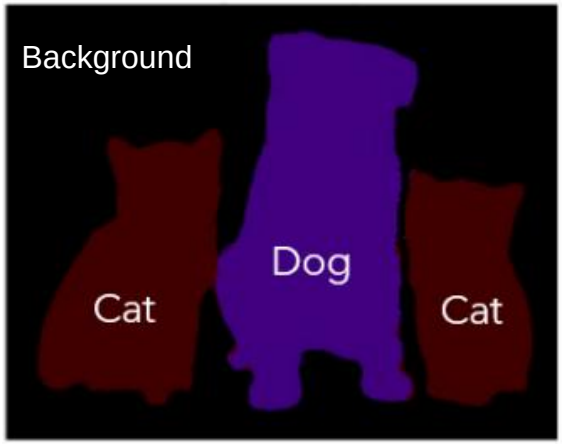
Input Image

network

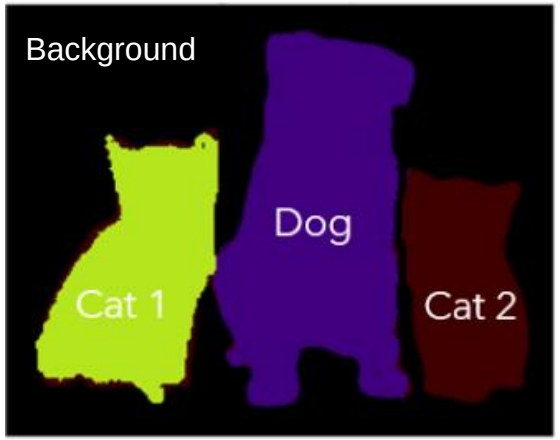
Object Detection



Semantic Segmentation



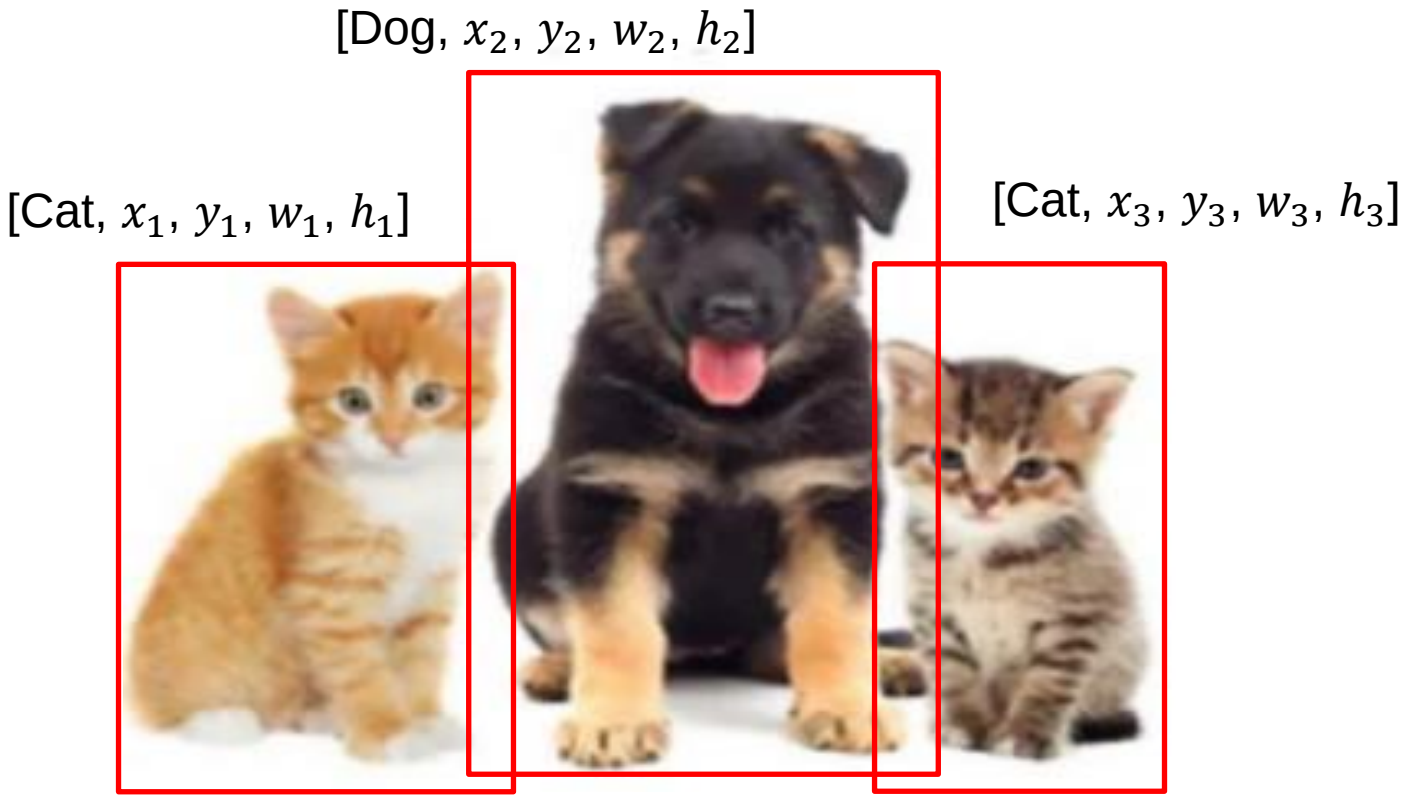
Instance Segmentation



Source: https://www.v7labs.com/blog/instance-segmentation-guide

Object Detection

- Also called „bounding box detection“
- Recognize the position and the class of each object by bounding box



$\boldsymbol{x}, \boldsymbol{y}$: origin of bounding box
$\boldsymbol{w}, \boldsymbol{h}$: width, height of bounding box

Semantic Segmentation

- Assign each pixel in the image with a category label
- Do not differentiate instances of the same category

Source: https://d2l.ai/chapter_computer-vision/semantic-segmentation-and-dataset.html

Instance Segmentation

- Each instance of a category will be assigned by a different label.

Semantic Segmentation

Instance Segmentation

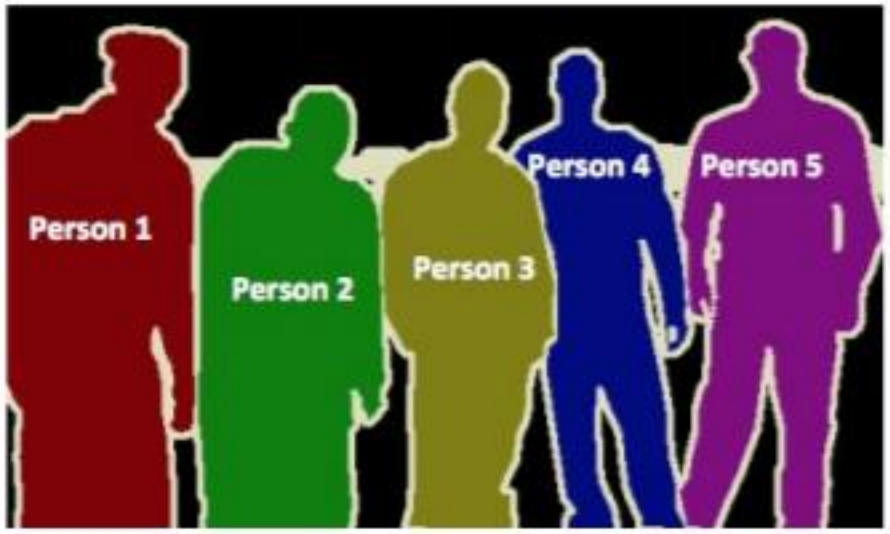


Image source: https://www.v7labs.com/blog/instance-segmentation-guide

Other Topics in Computer Vision

- Image Captioning
- Image Generation
- 3D Computer Vision
- Optical Flow
- ...

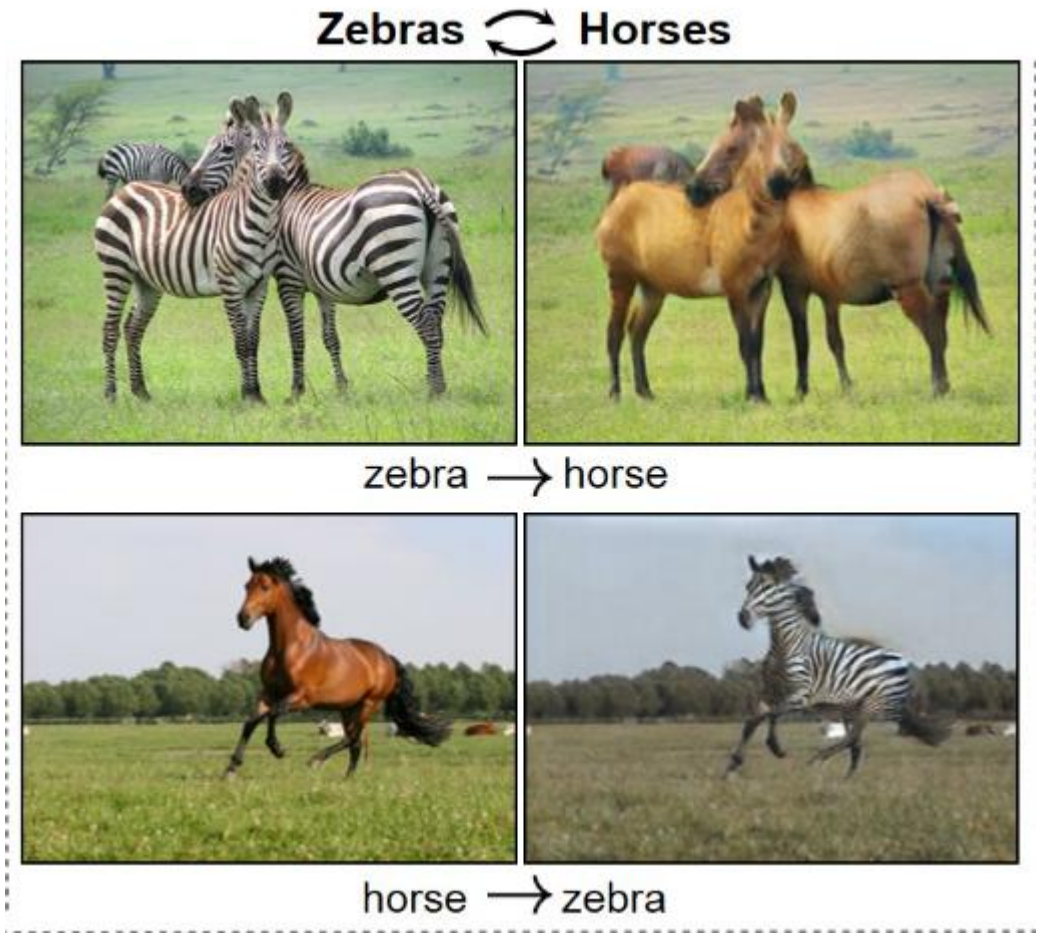
Image Captioning

“A man surfs on the sea”

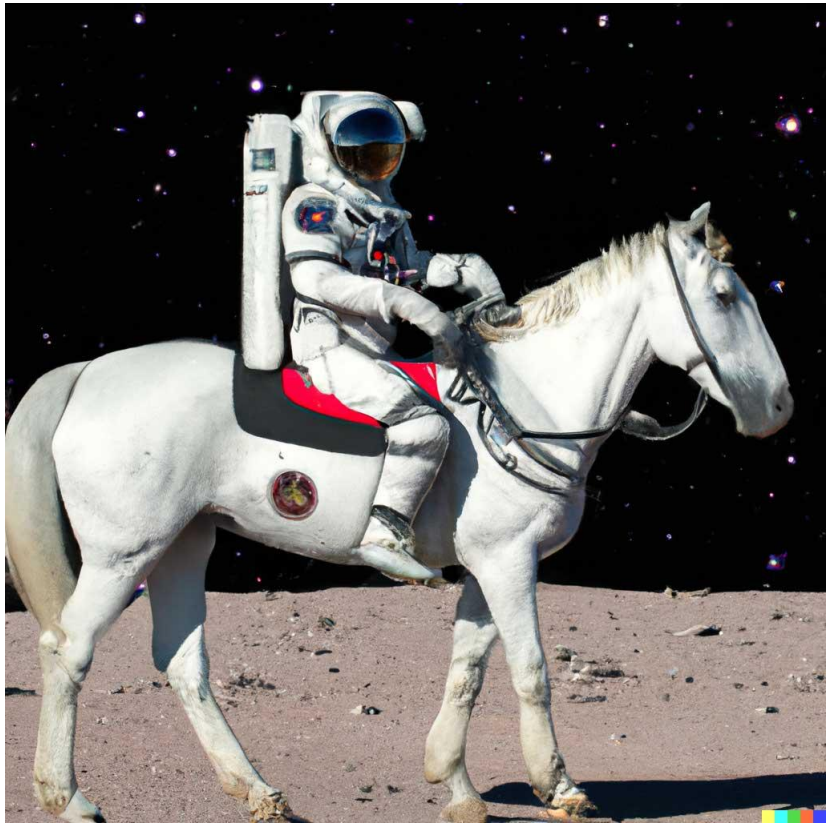
“Two dogs sit on the grass”

Image Generation

CycleGAN



DALL·E 2



Input: "An astronaut riding a horse in photorealistic style."

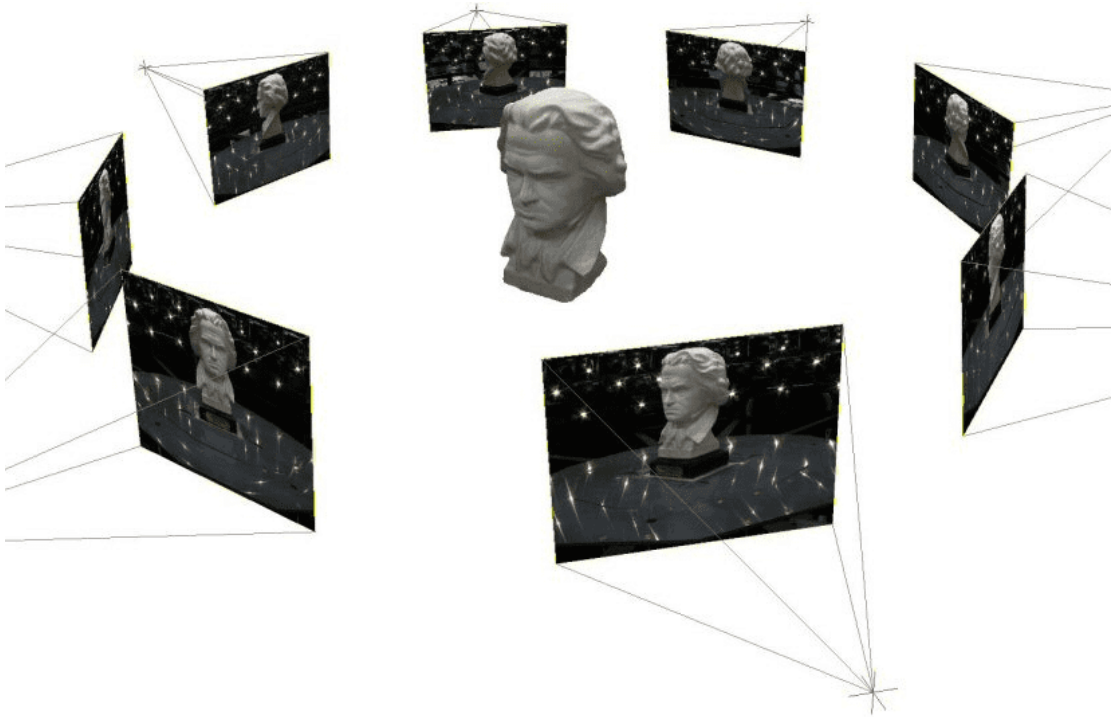
[1] Zhu, et. Al., Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV, 2017

3D Computer Vision

3D reconstruction from a single image

https://paperswithcode.com/task/3d-object-reconstruction-from-a-single-image/codeless

3D reconstruction from multiple images



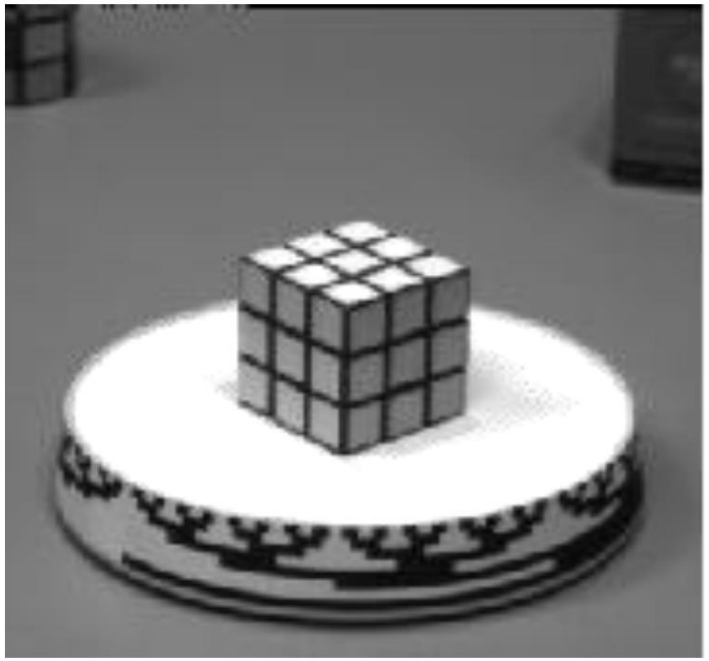
Method: Neural Radiance Fields (NERFs)

https://theaisummer.com/nerf/

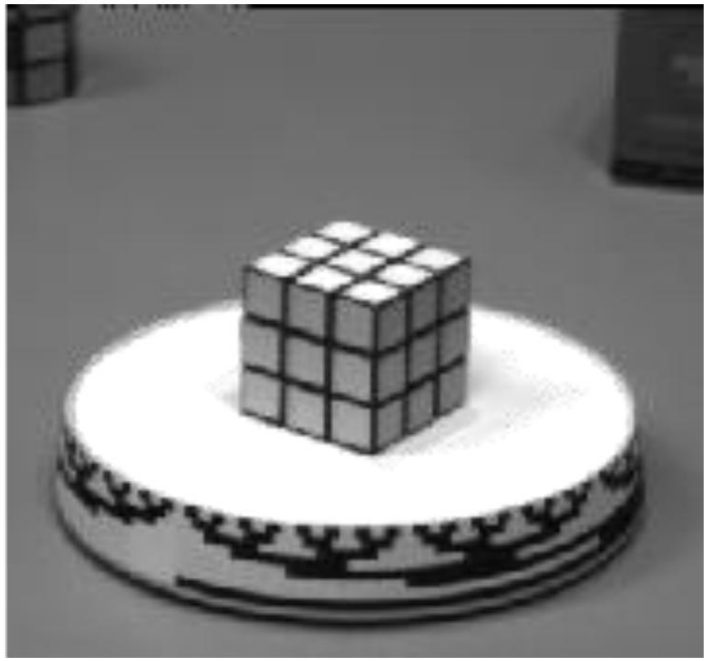
Optical Flow

- Estimate the movement of objects in single or multiple images

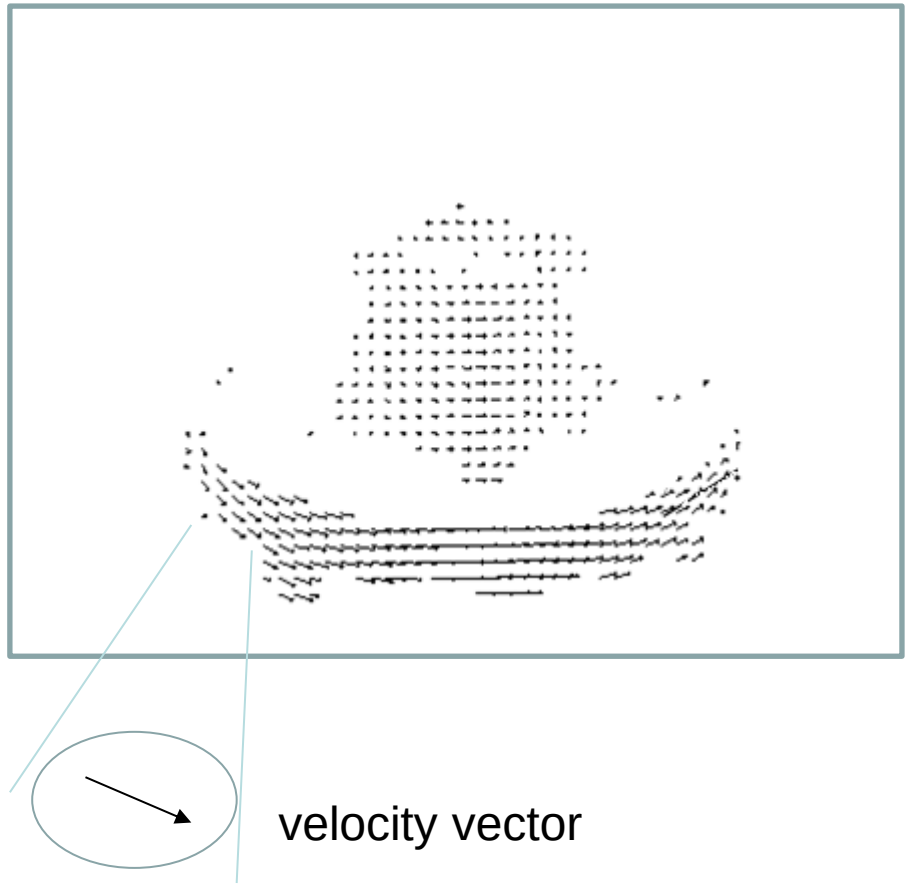
Time t_0



Time t_1



Optical flow



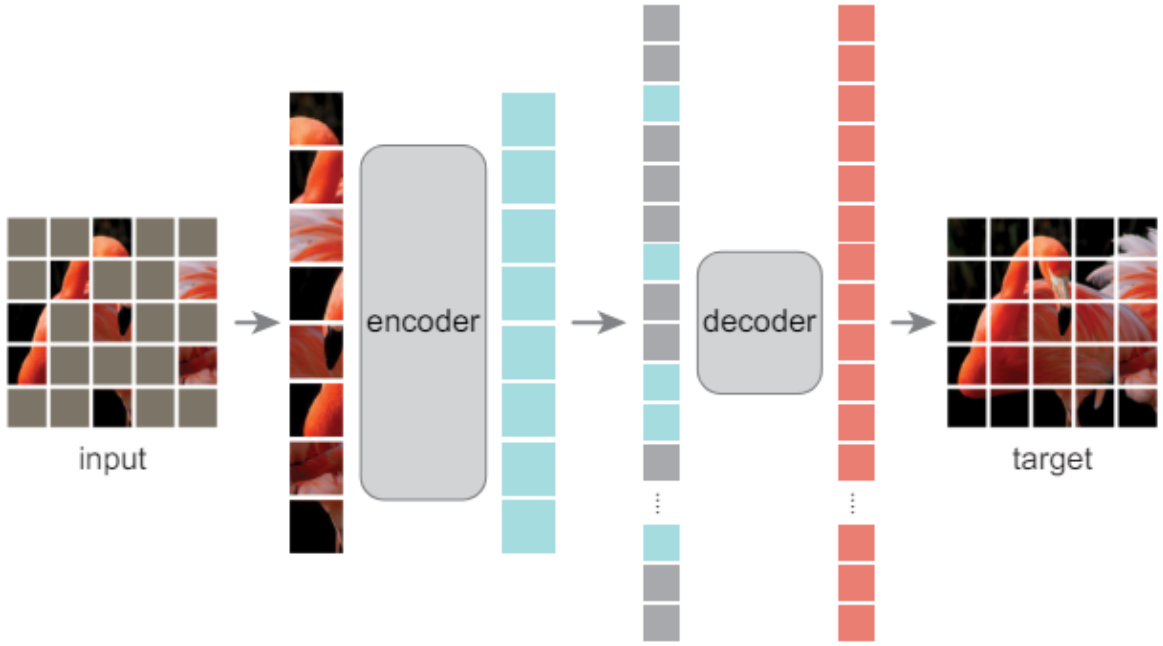
Optical Flow

Source: https://docs.opencv.org/3.4/d4/dee/tutorial_optical_flow.html

Modern advances in CV

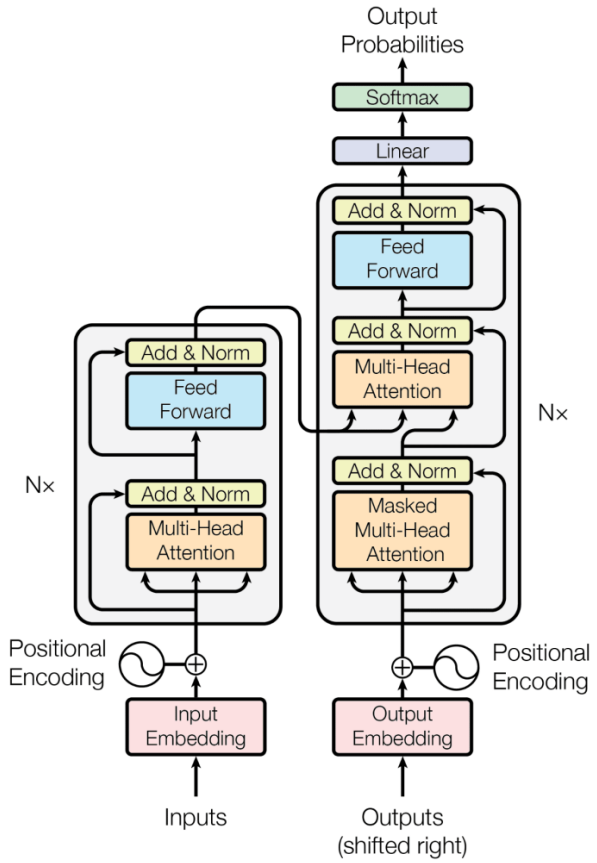
Transformation-based models

- ViT (Vision Transformer), 2021
- Swin Transformer, 2021
- MAE (masked autoencoder), 2021



MAE

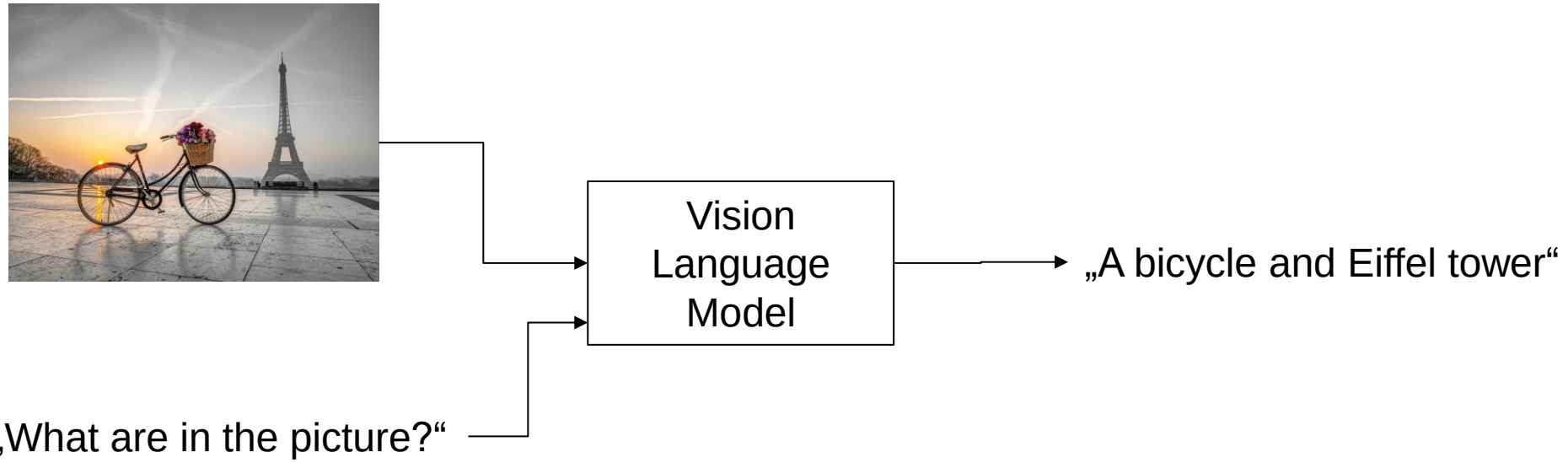
Transformer, 2017



[1] Liu, et.al., Swin Transformer: Hierarchical Vision Transformer using Shifted Windows, 2021
[2] He, et. al., Masked Autoencoders Are Scalable Vision Learners, aXiv, 2021

Vision Language Model (VLM)

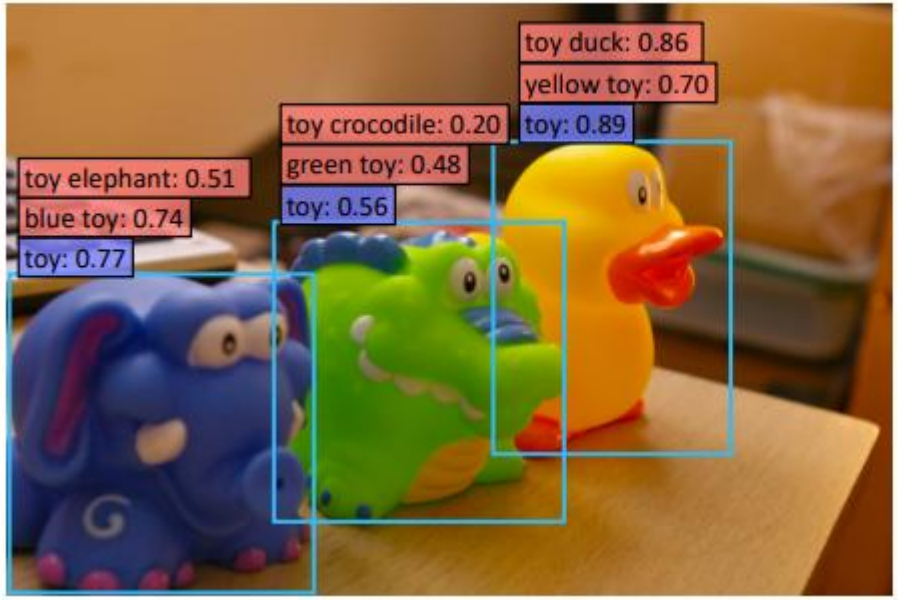
"A **vision-language model** is a fusion of vision and natural language models. It ingests images and their respective textual descriptions as inputs and learns to associate the knowledge from the two modalities." -- from Google

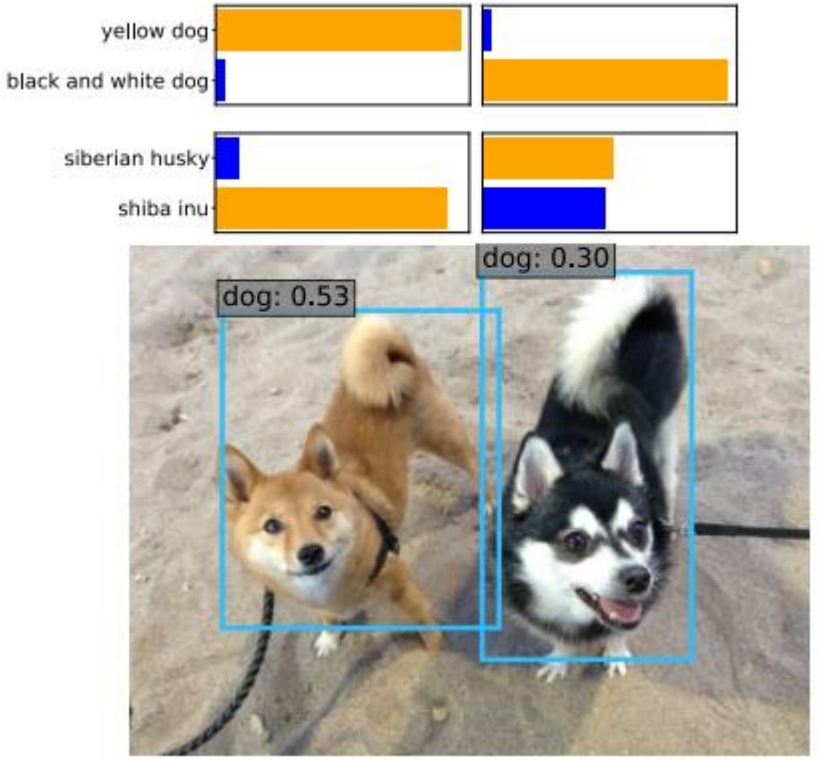


- Ideally, we want to ask any questions. But not all of them can be answered by VLM.

Open-Vocabulary Detection

- Detect objects according to **free text descriptions**, which may not have been seen during training

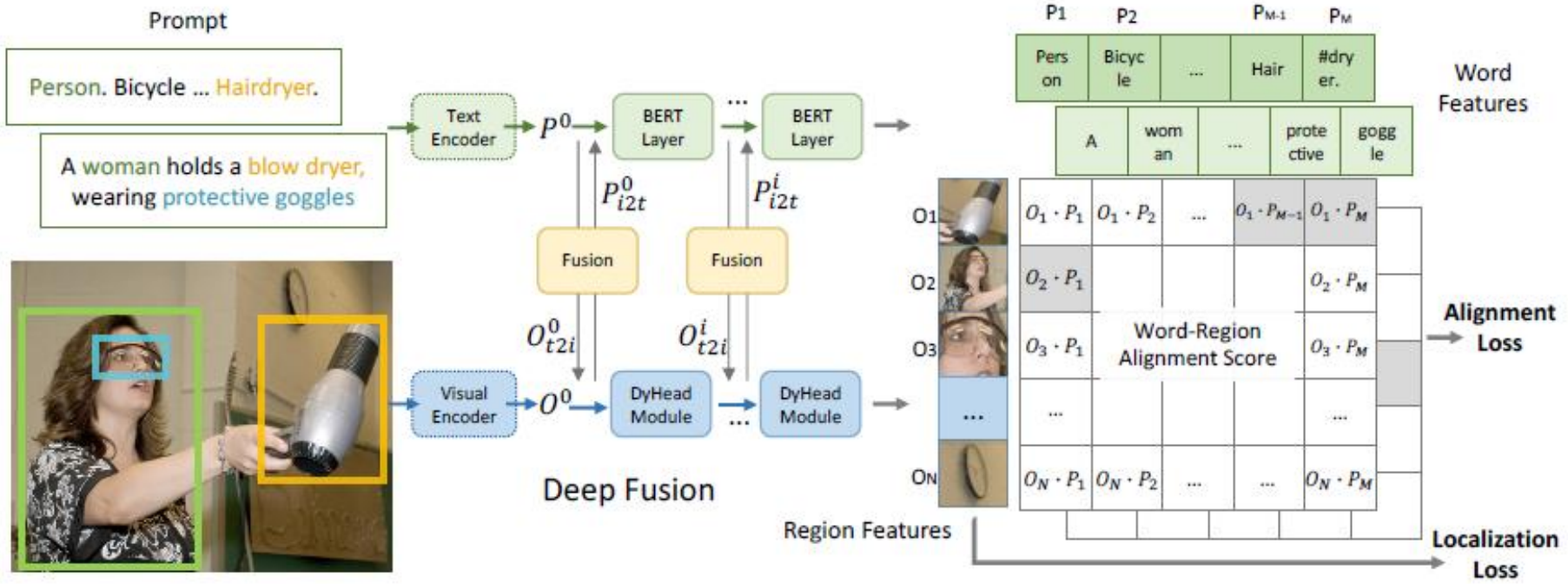


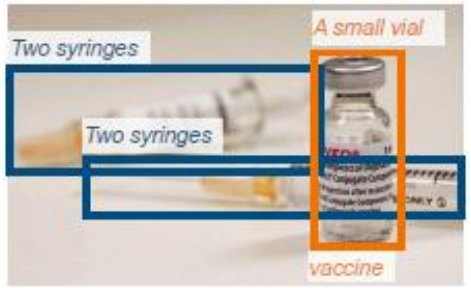


ViLD: Open-vocabulary detection via vision and language knowledge distillation, 2021

Open-Vocabulary Detection

- **GLIP**: Grounded Lange-Image Pre-training, 2022
- A unified framework for detection and grounding
- GLIP is accepted in CVRP 2022, won the Best Paper Award



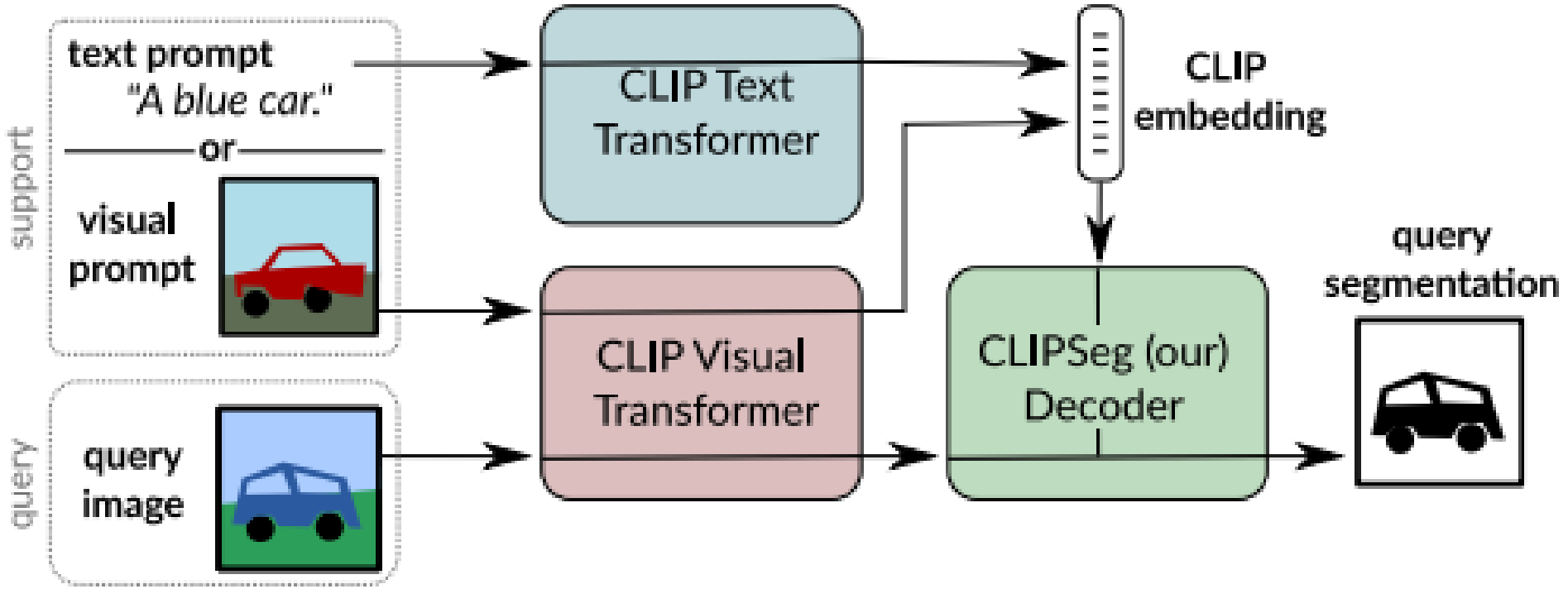


Two syringes and a small vial of vaccine.

playa esmeralda in holguin, cuba. the view from the top of the beach. beautiful caribbean sea turquoise

Open-Vocabulary Semantic Segmentation

- **ClipSeg**: Image segmentation using text and image prompts, 2022
  - Text prompt
  - Visual prompt



Other VLM …

- VL-BERT (Visual-Linguistic BERT), 2020
- ViLT (Vision-and-Language Transformer), 2021
- MDETR (text-conditioned object detection), 2021
- ALBEF (Align image and text before fusing), 2021
- SAM (point prompt), 2022
- X-Decoder, (open-vocabulary segmentation), 2023
- SegGPT (few shot learning), 2023

[1] Kim, et. al., ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision, arXiv, 2021
[2] Su, et. al., Vl-bert: Pre-training of generic visual-linguistic representations, ICLR, 2020
[3] Li, et. al., Align before Fuse: Vision and Language Representation Learning with Momentum Distillation, NeurIPS, 2021

2017

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com
Noam Shazeer* Google Brain noam@google.com
Niki Parmar* Google Research nikip@google.com
Jakob Uszkoreit* Google Research usz@google.com
Llion Jones* Google Research llion@google.com
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu
Łukasz Kaiser* Google Brain lukaszkaiser@google.com
Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

2025

„Vision Language Model is all you need“

Summary

- Where can you learn more?
  - Listed references
  - Search in Google