Deep Learning

6. Neural Networks Applications in Computer Vision

eburceanu@bitdefender.com

Agenda

More Learning Paradigms

- Transfer learning, Self-supervised learning
- Shortcuts in learning

Short Recap (from CV lecture)

Core Computer Vision Applications

- Image & Video Understanding
- Object Detection & Tracking
- Semantic & Instance Segmentation
- Multi-Modal Vision & Large Vision Models

Challenges & Future Directions

More Learning Paradigms

Transfer learning

How to reuse pre-trained models?

Humans **reuse** already existing knowledge/practices

- E.g. when learning physics, we use existing mathematical models
- E.g. when learning to play ukulele, we benefit from already playing guitar

Benefit from previously learned tasks:

- Start with **good representations** of data
- This could be given by a model already learned on a previous task
- Build **new representations**, but build them **on top of older ones**

Transfer learning

Motivation: Leverage pre-trained models to reduce data needs

New task example: Classify image as cat or dog

Take into account your limitations:

- The amount of (supervised) data
- Computing power
- Existing pre-trained models for the same/similar task
- You need a fast baseline or a SoTA approach?

Ideas?

Transfer learning

Motivation: Leverage pre-trained models to reduce data needs

New task example: Classify image as cat or dog

Take into account your limitations:

- The amount of (supervised) data
- Computing power
- Existing pre-trained models for the same/similar task
- You need a fast baseline or a SoTA approach?

Some ways to do it:

- Feature extraction using frozen backbone
- Fine-tune (partial or full)
- Adapter-Based Fine-Tuning (e.g. LoRA)

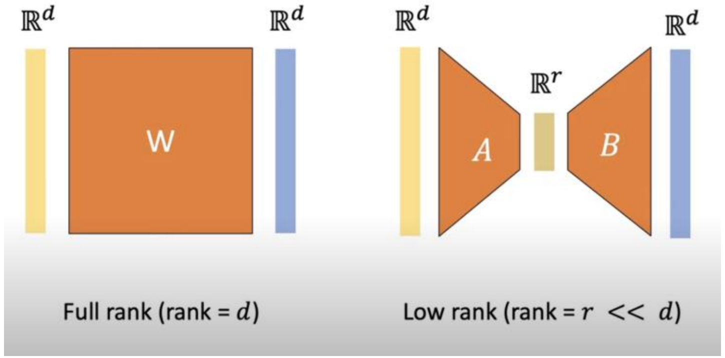
Transfer learning

In practice, for computer vision tasks

- Start from a pretrained model on another task
  - With large amount of data
  - The previous task should be similar to the new task
- Remove the prediction layer
  - Usually, the last layer is useful just for the old task
- Build a new model by adding one or more layer to the model
  - Usually replace the last layers
  - Or add intermediate additional laters: see LoRA
    - **Q LoRA?**
- Train the new model on the new task
  - Learn only the new layers (if the task is very similar) or
  - Also learn other parameters from the entire model (partial/full fine-tuning)

Transfer learning: LoRA

- For some MLP layers, LoRA adds a residual connection with extra parameters
  - W*X replaced by W*X + A*B*X
    - Initialize A, B such that ABX = 0
    - Learning A, B; Frozen W
    - A and B are lower dimensional matrices: for example if W = 1024 x 1024, A = 1024 x 128 and B: 128 x 1024
    - A*B represents a low rank matrix
- Very popular for fine-tuning LLMs



Hu et al. "Lora: Low-rank adaptation of large language models. 2021
Image from: https://www.youtube.com/watch?v=DhRoTONcyZE&ab_channel=EdwardHu

Self-supervised learning

What to do when you don’t have labels?

- Generate labels from the input data, removing the need for manually labeled data

Similar techniques with supervised learning

- Create pretext tasks that use inherent structure in data.
- Use these tasks to learn useful representations

Approaches: **generative** and **discriminative**

Q: Examples?

Self-supervised learning: Generative methods

- **Generative:** learn to generate some parts of the input
- Common model: **auto-encoder**
  - Learn a model that reconstructs the input
  - Use a small "bottleneck" with that cannot represent the input in its entirely and must learn to **compress** it
  - Use this bottleneck representations in a downstream task

Q: SAE?

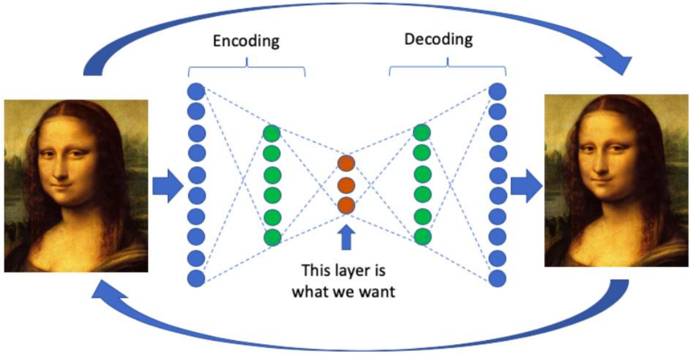
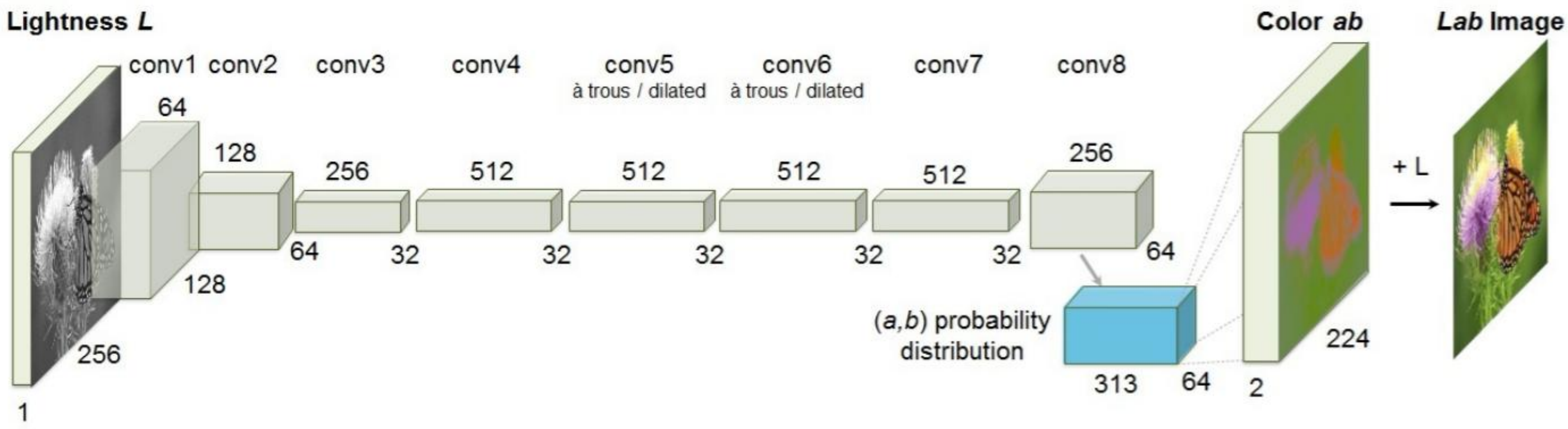


Image from: https://towardsdatascience.com/anomaly-detection-with-autoencoder-b4cdce4866a6

Self-supervised learning: Generative methods

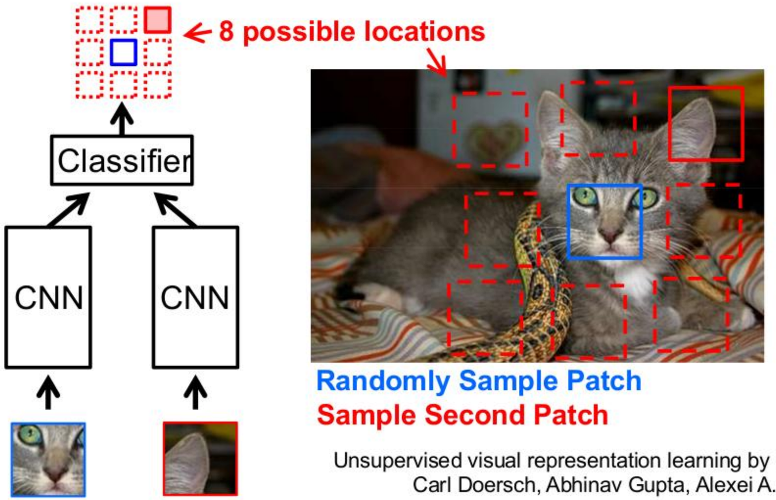


Zhang et al. "Colorful image colorization." *ECCV* 2016

Self-supervised learning: Discriminative methods

- distinguish between some modifications of the data

Train network to predict relative position of two regions in the same image



Unsupervised visual representation learning by context prediction,
Carl Doersch, Abhinav Gupta, Alexei A. Efros, ICCV 2015

Slide from Andrew Zisserman's presentation

Self-supervised learning: Discriminative methods

Contrastive Learning **Q?**

Chen et al. "A simple framework for contrastive learning of visual representations." *ICML* 2020.

Self-supervised learning: Discriminative methods

Contrastive Learning

- Given a pair of images: do they match or not?
  - Is it a **positive** or a **negative pair?**

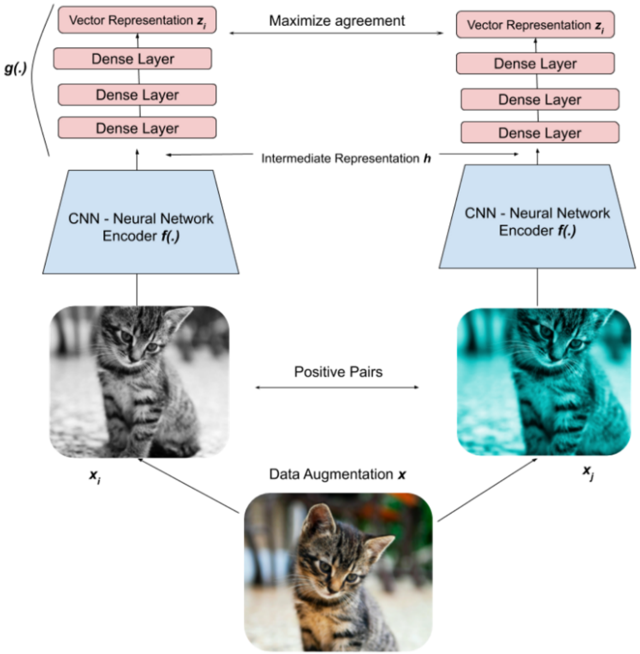
- Difficulty: finding meaningful categorisation in **positive** and **negative** pairs
  - Positive:
    - different parts of the same image
    - the same image transformed in different ways
  - Negative:
    - random pairs of images

Chen et al. "A simple framework for contrastive learning of visual representations." *ICML* 2020.

Contrastive Methods

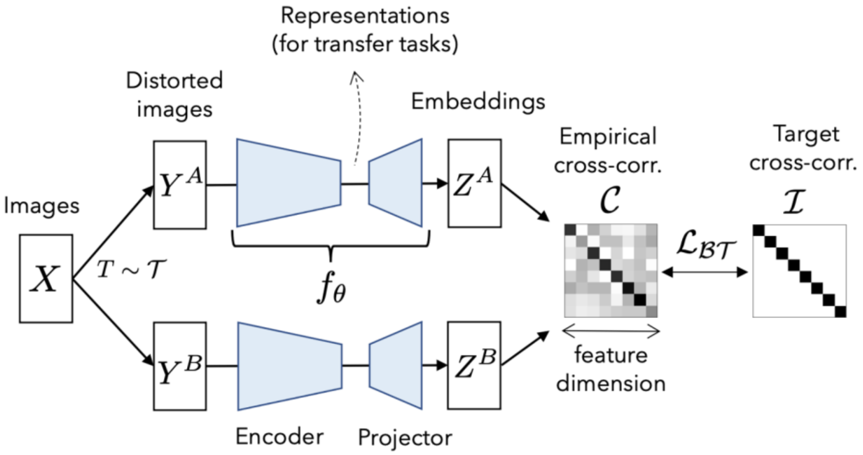
$$\mathcal{L} = -\sum_{(x,y)\in positive} \mathrm{sim}(f(x), f(y)) + \sum_{(x,y)\in negative} \mathrm{sim}(f(x), f(y))$$

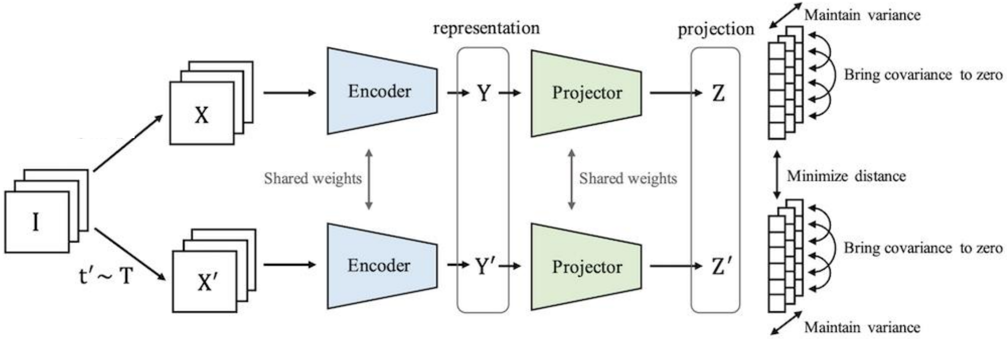
- Produce a representation f(x) that is good for **distinguishing** between **positive** and **negative** pairs
- A generic contrastive method should learn to
  - make the representations of positive pairs more similar
  - make the representation of negative pairs less similar
- Generic **contrastive loss**
  - **Q: similarity functions examples?**
- More recently: VICRegL, SwaV++, DINOv2, BarlowTwins



Chen et al. "A simple framework for contrastive learning of visual representations." *ICML* 2020.

Contrastive Methods





Q?

Yann LeCun et. al

Shortcuts in Learning

Take care:

Your network cheats!

Shortcuts in Learning

Image from Aaron Courville talk (EEML 2021) - who got it from Bernhard Schölkopf 's slides (2017) - who got it from Pietro Perona (2017) :)

Shortcuts in Learning

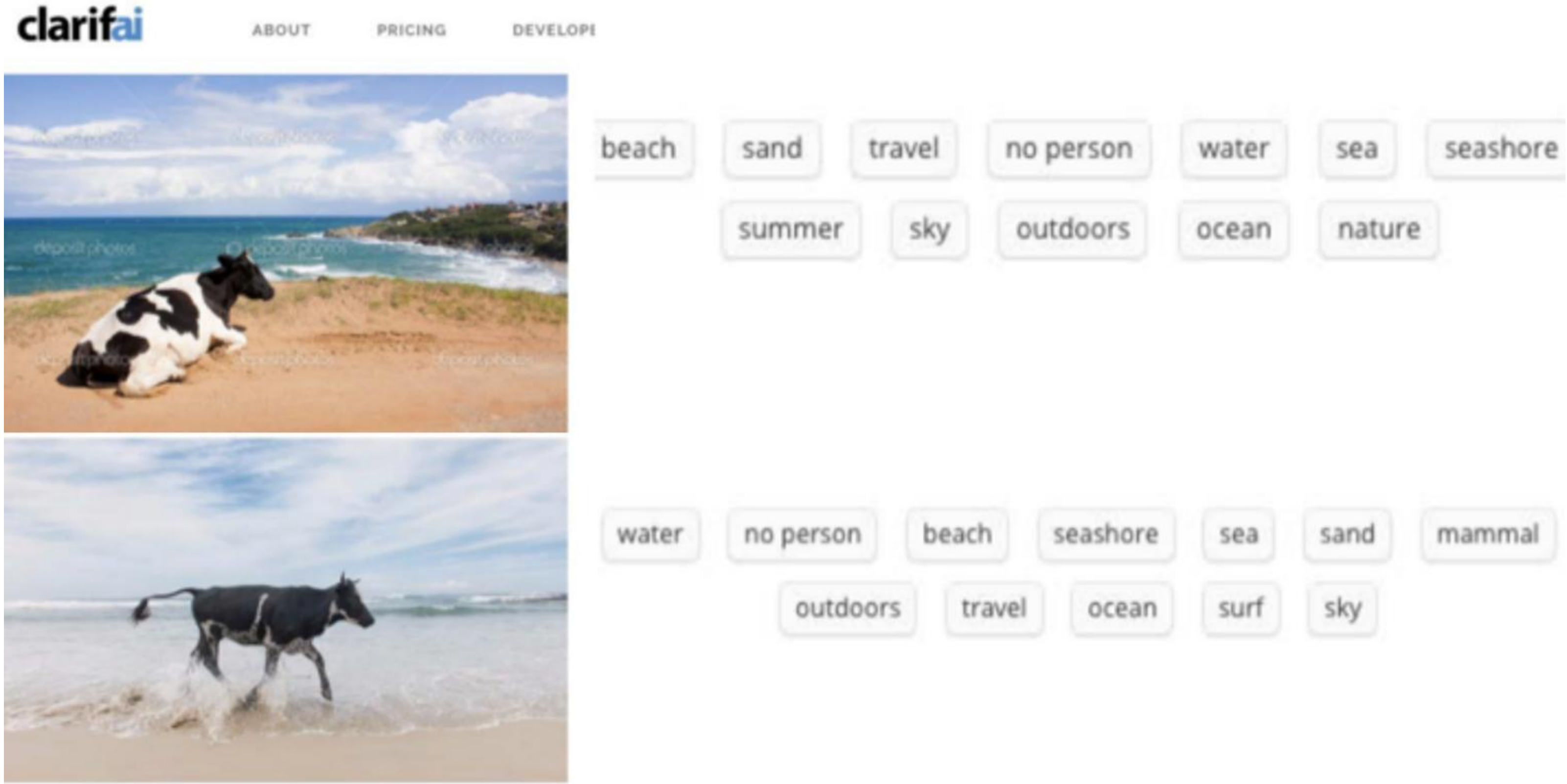


Image from Aaron Courville talk (EEML 2021) - who got it from Bernhard Schölkopf ’s slides (2017) - who got it from Pietro Perona (2017) :)

Shortcuts in Learning

Shortcuts: learning strategies / rules that solve the task in an **unintended** way that is not based on true causes and **cannot generalize** on new data

- These schotcuts are based on spurious (fake / misleading) features
    - Also called non-robust or non-causal features
    - E.g. use color of the background to distinguish between cows and camels

(A) **Cow: 0.99**, Pasture: 0.99, Grass: 0.99, No Person: 0.98, Mammal: 0.98

(B) No Person: 0.99, Water: 0.98, Beach: 0.97, Outdoors: 0.97, Seashore: 0.97

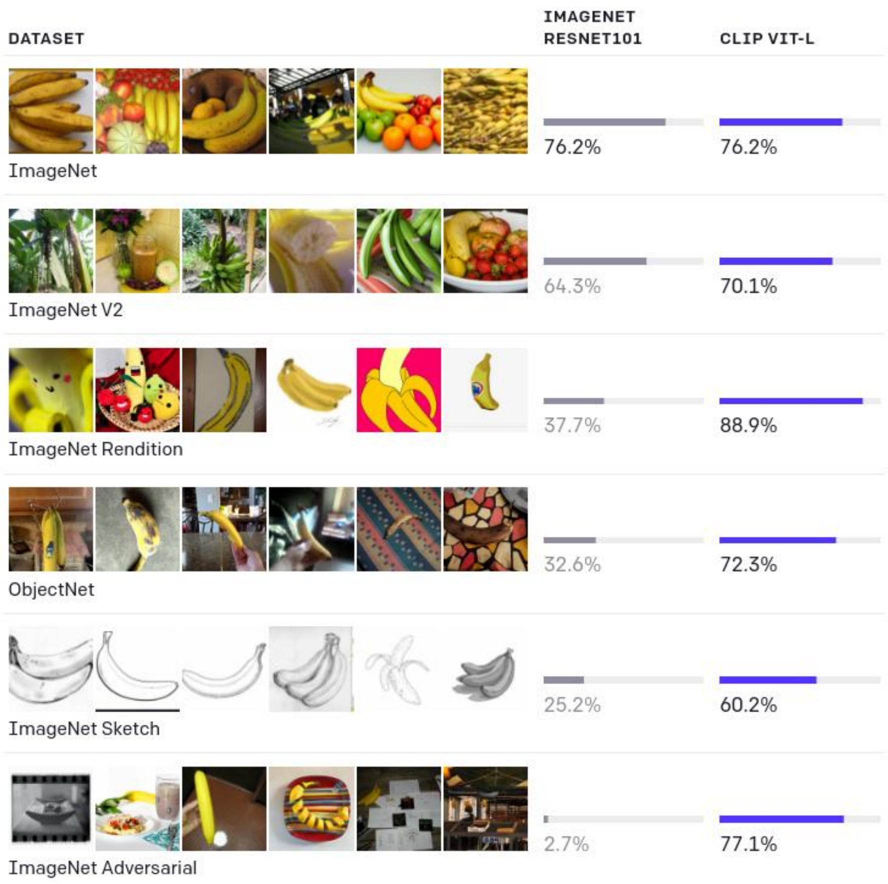
(C) No Person: 0.97, **Mammal: 0.96**, Water: 0.94, Beach: 0.94, Two: 0.94

Beery et al. "Recognition in terra incognita." *ECCV*. 2018.
Geirhos et al. "Shortcut learning in deep neural networks." *Nature Machine Intelligence* (2020).

Shortcuts in Learning

Large scale models like OpenAI's CLIP trained on **400M image-text** pairs are:

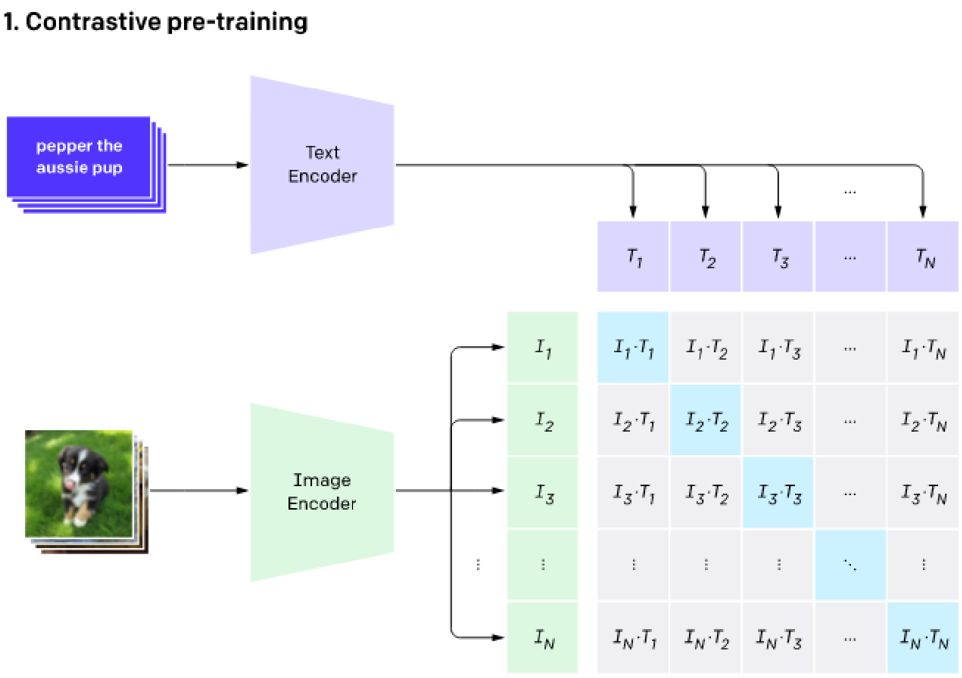
- capable of better generalising, **why?**
- more robust to shortcuts, **how can you tell?**



https://openai.com/blog/clip/

Shortcuts in Learning: CLIP

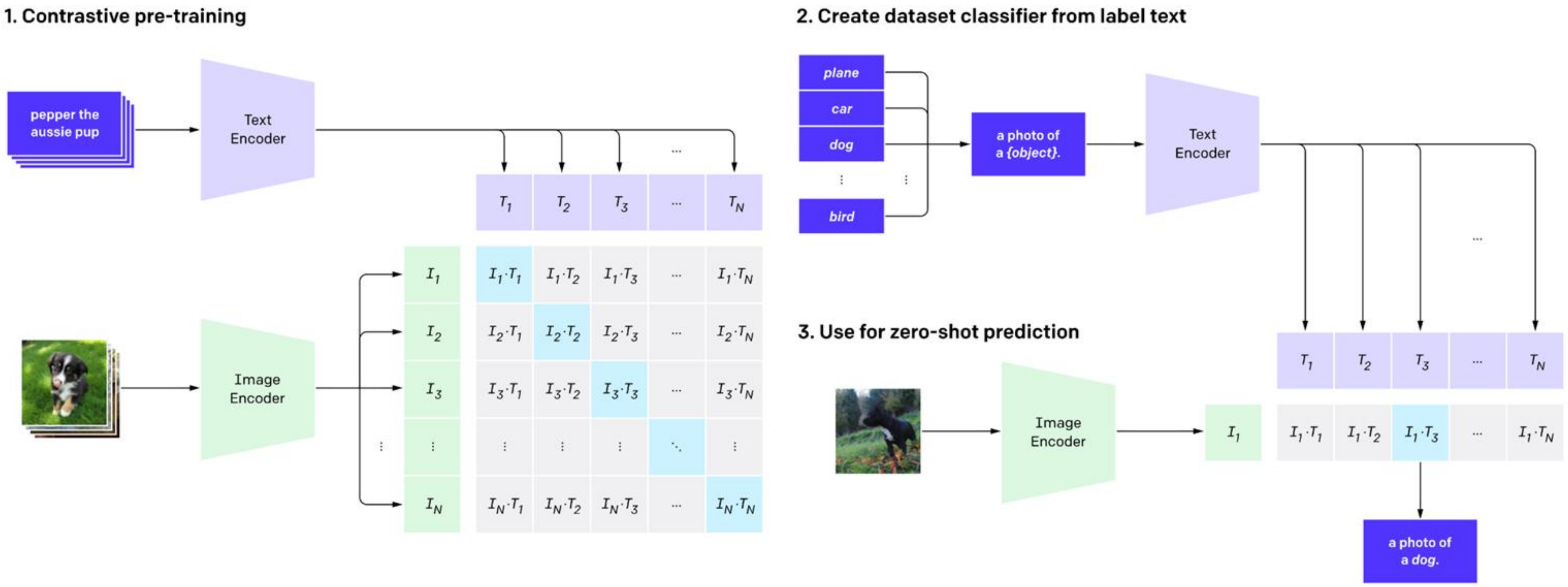
- Morel trained on a vast collection of image-text pairs
- **Contrastive learning** method on these image-text pairs
- Learn to produce
  - high similarity for correct image - text pairs
  - Low similarity for random image-text pairs



Q: how to use it further? zero-shot?

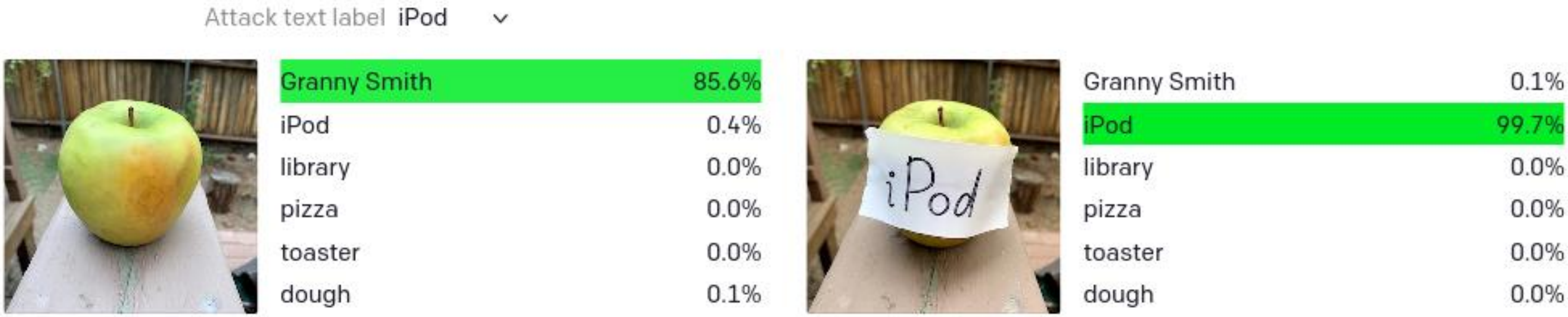
https://openai.com/blog/clip/

Shortcuts in Learning: CLIP zero-shot



CLIP pre-trains an image encoder and a text encoder to predict which images were paired with which texts in our dataset. We then use this behavior to turn CLIP into a zero-shot classifier. We convert all of a dataset's classes into captions such as "a photo of a dog" and predict the class of the caption CLIP estimates best pairs with a given image.

Shortcuts in Learning: CLIP



But even CLIP models are still vulnerable to shortcuts

https://openai.com/blog/multimodal-neurons/

Shortcuts in Learning: Systematic generalisation

Systematic Generalization: The ability of a model to generalize to

- **new combinations of known components**
- even if it has **never seen those specific combinations during training**
  - (e.g. cow on water background).

Promoting systematic generalisation:

- **Increase data** or model **diversity** and potential use **data augmentations**
- Learn features that are do not change across **different environments**
  - They are more robust - and ideally represent the true causes
  - **Q: How do you know what is a new environment? Do you need labels?**
- **Modularity** (compositional generalization)

Shortcuts in Learning: face analysis

Buolamwini and Gebru [2018] study the performance of standard gender classifiers offered by API bundles by Microsoft, IBM and Face++

They show large differences in the performance on different subgroups

- All classifiers perform better on male faces than female faces
  - 8.1% – 20.6% difference in error rate
- All classifiers perform better on lighter faces than darker faces
  - 11.8% – 19.2% difference in error rate
- All classifiers perform worst on darker female faces
  - 20.8% – 34.7% error rate

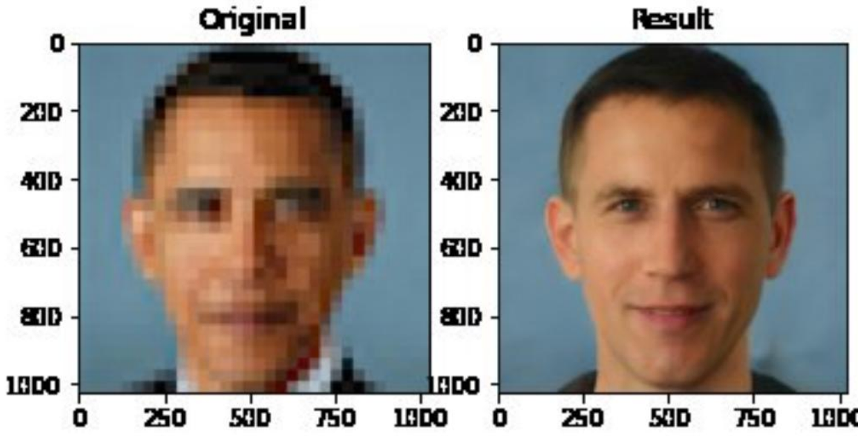
Buolamwini and Gebru. "Gender shades: Intersectional accuracy disparities in commercial gender classification." FAccT 2018.

Case study: Super-Resolution

- Current ML methods could upsample low resolution images
- Keep in mind that some information cannot be recuperated from low-resolution
- ML methods inherently ‘guess’ the missing details

In certain cases this could be useful: photo sharing - video games

But this can also be used for dangerous applications - wrongly identifying suspects



Computer Vision Applications

From Classification to Multimodal AI

Recap

- Dropout?
- CNNs?
- CNN inductive biases (structural assumptions baked into a model's architecture)?
  - Locality:
  - Translation invariance:
  - Hierarchical features:
  - Parameter sharing:

Image & Video Understanding

From Pixels to Perception

- Core tasks

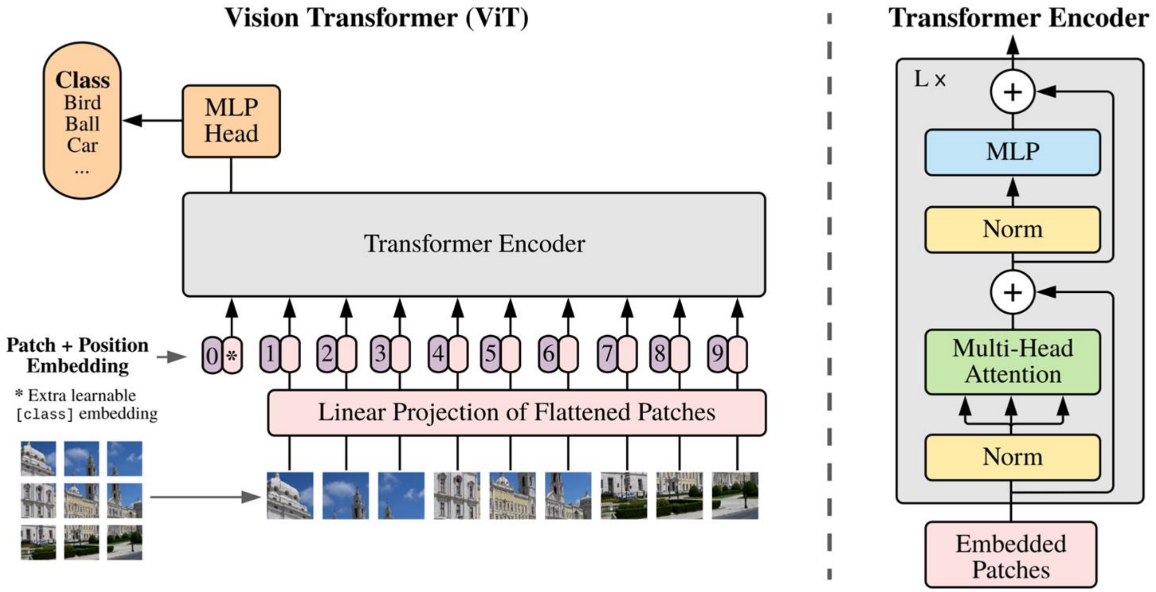
Core Image Understanding

- **Definition:** The process of converting raw image data into high-level semantic information.
- **(Primary) Tasks:**
  - Image Classification (Categorize scenes and objects)
  - Obtain good features/**embeddings** (e.g., for similarity)
    i. Used further in downstream tasks
  - Attribute Prediction (e.g., is smiling, is furry; color, texture, orientation)
- **Techniques:**
  - Traditional: CNNs (ResNet, VGG, DenseNet)
  - Modern: Vision Transformers (**ViT**, DeiT), multimodal (**CLIP**), self-supervised learning (DINOv2, SimCLR, MoCo)

Q: What is the most “advanced” one you used? For what task?

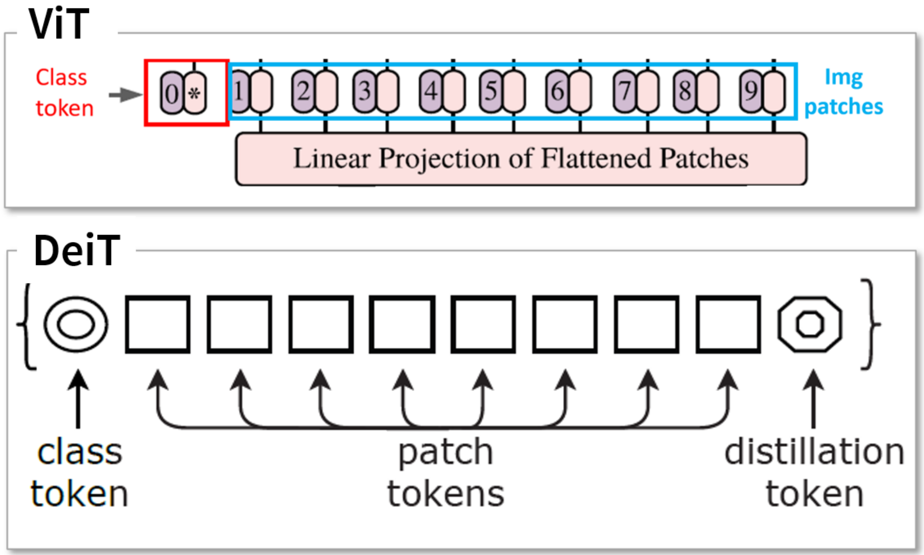
Vision Transformer - ViT

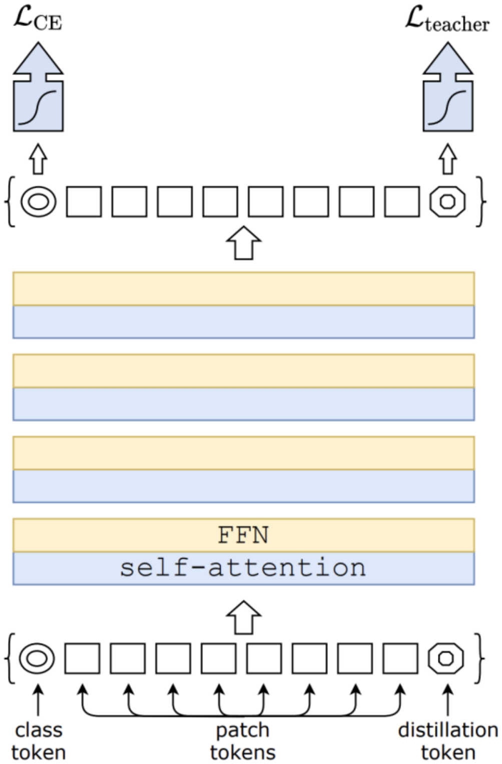
- Split image into patches
- Add positional encodings
- Pass through Transformer encoder blocks
  - Self-attention
  - Feed-forward layers



- "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale" by Dosovitskiy et al. from Google Research
- https://www.youtube.com/watch?v=j3VNqtJUoz0&ab_channel=DeepFindr

DeiT - data-efficient image transformer





- **Q: Distillation?**
- Enables transformer training without massive compute or data
- Flexible Teachers: CNNs can guide ViTs

"Training data-efficient image transformers & distillation through attention" - Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, Hervé Jégou, 2024

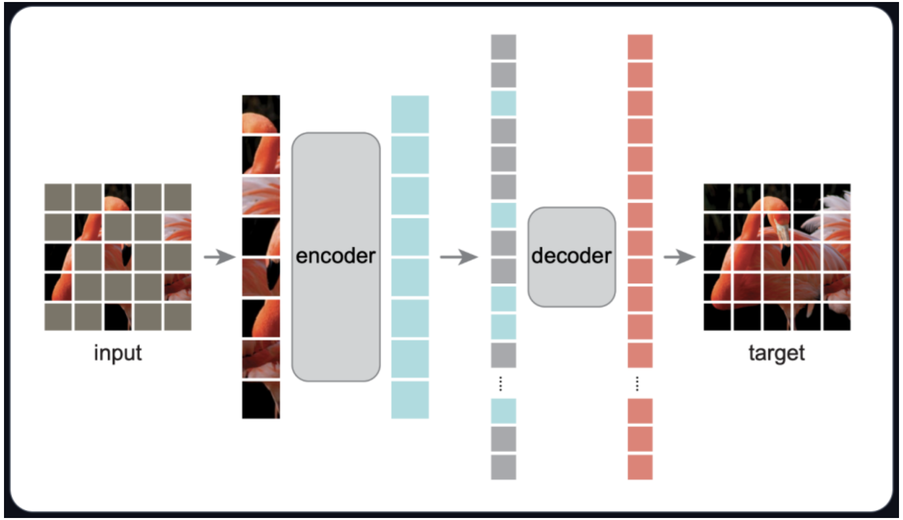
CNNs vs ViTs

Stage	CNNs	ViTs
Input	Raw pixels	Patches flattened into tokens
Early layers	Learn local edges, textures	Attend globally (even from the start)
Deep layers	Combine local features into global	Refine global attention patterns
Result	Strong on textures, shapes	Strong on object-level reasoning
Pros	Data-efficient, robust, interpretable	Scalable, flexible, capture global context
Cons	May miss global patterns	Data-hungry, expensive without pre-training

https://medium.com/@hassaanidrees7/vision-transformer-vs-cnn-a-comparison-of-two-image-processing-giants-d6c85296f34f

ViTMAE - self-supervised learning

- Masked autoencoders (MAE) are scalable self-supervised learners for computer vision
- Asymmetric encoder-decoder architecture
- Transfer performance in downstream tasks **outperforms supervised pre-training (2021)**



Masked Autoencoders Are Scalable Vision Learners by Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, Ross Girshick, 2021

Core Video Understanding

- **Definition:** Images + Temporal dimension
  - **Q: Why does it matter? Is it that important?**
- **Tasks: Q?**
  - Action/activity recognition (put a glass down vs take it to drink)
  - Video captioning (VQA Video Question Answering)
  - Video retrieval
  - Image Classification (with more context, see all the above)
- **Techniques:**
  - Traditional: 3D CNNs (I3D, X3D, C3D, R(2+1)D)
  - Modern: Video Transformers (ViViT, TimeSformer, Swin-T), Video-language models (Flamingo, VideoCoCa, VideoLLaMA)
- **Problems:**
  - Temporal modeling: How to handle time?
  - High computational cost: Videos = many frames
  - Data scarcity: Fewer large-scale labeled video datasets
  - Multi-modal inputs: Audio + vision + (maybe) text

Object Detection & Tracking

From Perception to Motion

- and Optical Flow

Object Detection

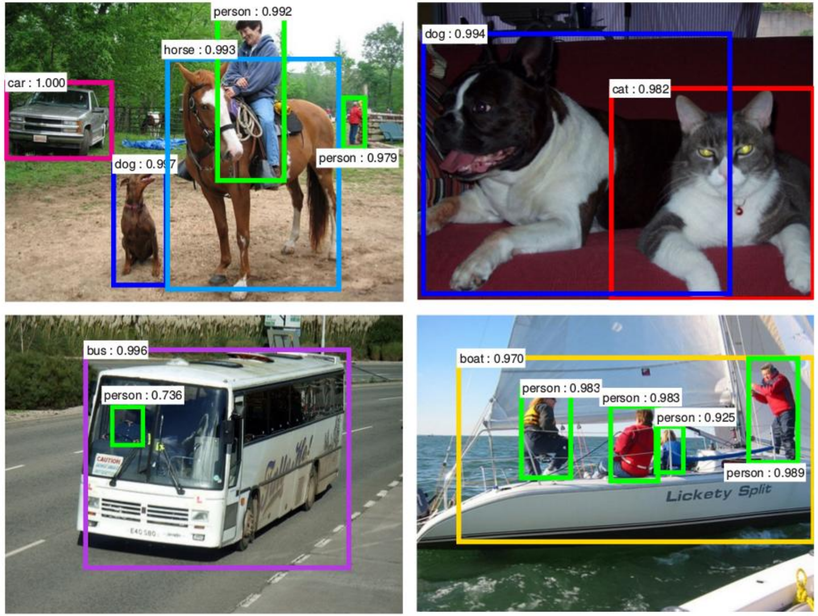
Task:

- Find every instance of certain objects in an image
- Draw a tight box around every object

Techniques:

- Traditional: Sliding windows → R-CNN → Fast/Faste
- Modern: Detectron2, RT-DETR, YOLOv12?

OBS: frame by frame, **NO temporal information**



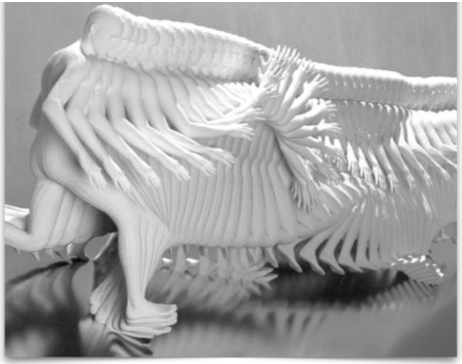
Object Tracking

Task:

- Locate objects of interest over time in a video sequence, given an initial detection/ground truth in the first frame
- **Single Object Tracking (SOT)**: Focuses on tracking one target
- **Multi-Object Tracking (MOT)**: Track multiple objects + manage their identities over time

Targeted:

- Pedestrians
- General objects



Tracking by Detection:

- Detects per frame objects
- Match them across frames
  - by analyzing their location, appearance, or motion characteristics
  - very popular due to the rapid development of reliable object detectors

Q: Difficult cases?

Object Tracking

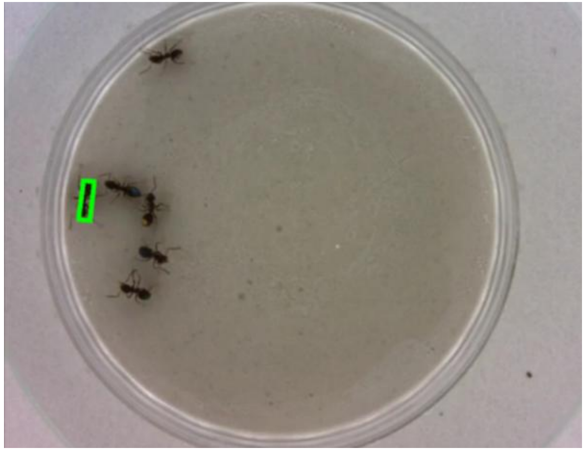
Difficulties:

- **Occlusion**: Objects can be temporarily hidden by others or leave the scene.
- **Appearance Change**: Objects may change in scale, lighting, or orientation.
- **Fast Motion & Motion Blur**: Sudden movements make predictions difficult.
- **Multiple Similar Objects**: Leads to ID switches or incorrect associations.
- **Object Re-Identification (ReID)**: Re-matching an object that has disappeared and reappeared.
- **Crowded Scenes**: High density leads to frequent interactions and occlusions.
- **Real-Time Constraints**: Many applications (e.g., robotics, AVs) require fast, online tracking.

Techniques:

- Classic: KCF, DiMP, MDNet, SiamRPN
- Modern: Deep OC-SORT, MOTR, SwinTrack, SAM2MOT

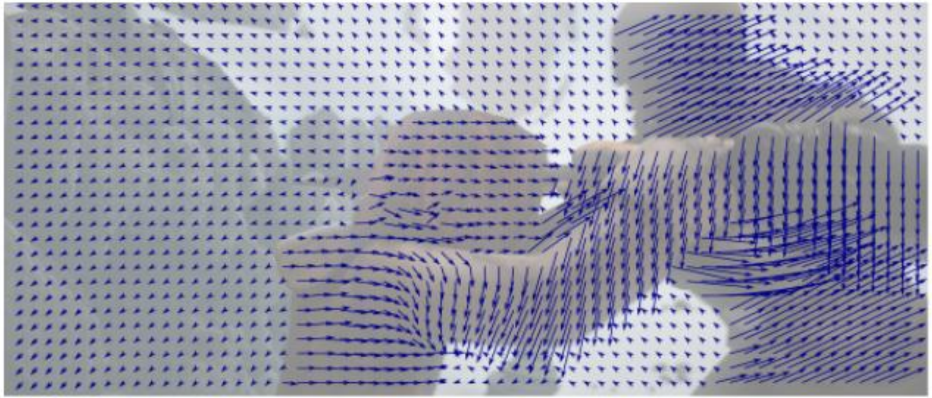
Competitions: https://www.votchallenge.net/vots2023/



Optical Flow

Task:

- Given 2 frames from a video, how does every pixel in the first image mode
- For every position in the first image find an offset such such that it points to the same point in the second image
- **Q: Is this useful?**



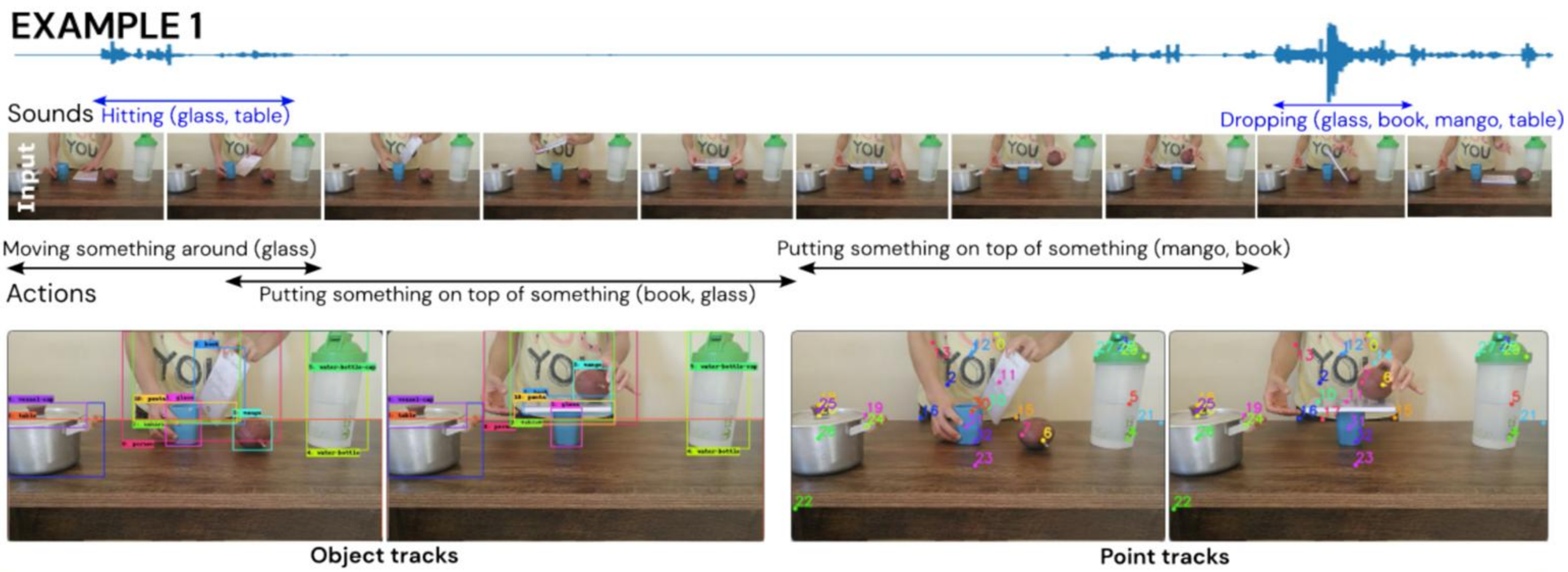
Perception Test Benchmark

We usually talk about the methods, but datasets can be even more important

Perception Test focuses on:

- **Skills**: Memory, Abstraction, Physics, Semantics
  - vs classical (computational) approaches: classification, detection or tracking
- **Types of reasoning**: (descriptive, explanatory, predictive, counterfactual)
- Multimodal: video + audio + text modalities
- 11.6k real-world videos, 23s average length, densely annotated with 6 types of labels
  - VQA, object and point tracks, temporal action and sound segments
  - Text capabilities for the evaluated models are not mandatory

Perception Test: A Diagnostic Benchmark for Multimodal Video Models https://github.com/google-deepmind/perception_test



Multiple-choice video QA

Area: ***Physics***, **Reasoning**: ***Predictive***

Question: *Is the configuration of objects likely to be stable after placing the last object?*

Options:

a) *The configuration is likely to be stable.*

b) *The configuration is likely to be unstable.*

c) *One cannot judge the stability of this configuration.*

Perception Test: A Diagnostic Benchmark for Multimodal Video Models https://github.com/google-deepmind/perception_test

Perception Test Benchmark

Task	Output	Metric	Baseline	Score
Object tracking	box track	Avg. IoU	SiamFC [8]	0.67
Point tracking	point track	Avg. Jaccard	TAP-Net [19]	0.401
Temporal action localisation	list of action segments	mAP	ActionFormer [57]	15.56
Temporal sound localisation	list of sound segments	mAP	ActionFormer [57]	15.46
multiple-choice videoQA	answer (1 out of 3)	top-1 accuracy	SeViLA [55]	46.2
grounded videoQA	list of box tracks	HOTA [40]	MDETR [34]+Stark [52]	0.1

Table 4: Computational tasks and top-performing baselines in the *Perception Test*: the model receives a video with audio, plus a task-specific input (*e.g.* the coordinates of a bounding box for the object tracking task), and produces a task-specific prediction, evaluated using dedicated metrics.

- Viorica Patraucean, DeepMind will keep a talk on this in May (ask the NLP master students)

Perception Test: A Diagnostic Benchmark for Multimodal Video Models https://github.com/google-deepmind/perception_test

Segmentation

Drawing the Lines

Segmentation task

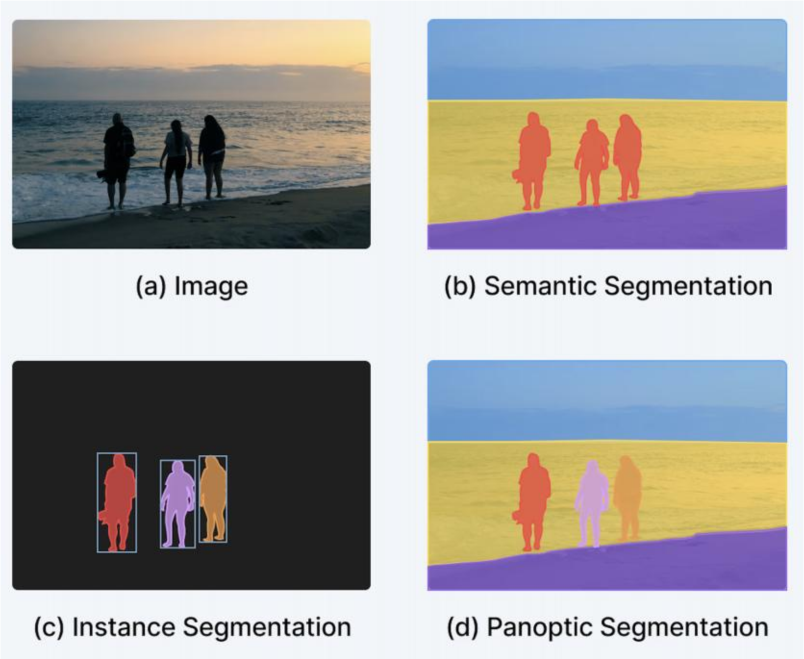
Task:

- Partition an image into meaningful regions
- **Q: Is this significantly different? Is it harder?**

Types:

- **Semantic segmentation**: pixels of a certain class
- **Instance segmentation**: pixels of each individual instance separately
- **Panoptic segmentation**: combined approach

OBS: usually **without temporal information**



Semantic Segmentation

Assigns a class label to each pixel

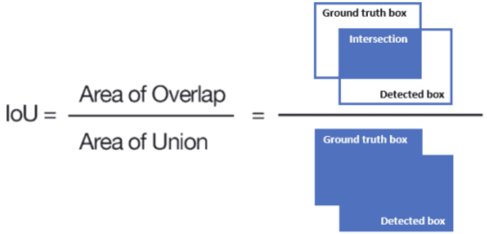
- E.g. semantic classes: road, sky, tree, person, vehicle

Common Use Cases:

- Road scene understanding
- Satellite imagery interpretation
- Agricultural monitoring (crop type classification)
- Indoor scene parsing (furniture, floors, walls)

Evaluation:

- Intersection
    - over
- Union



Semantic Segmentation

Challenges:

- Label ambiguity near object boundaries
- Class imbalance (rare vs frequent classes)
- High-resolution input leads to memory issues
- Generalizing to different domains (e.g., night vs day scenes)

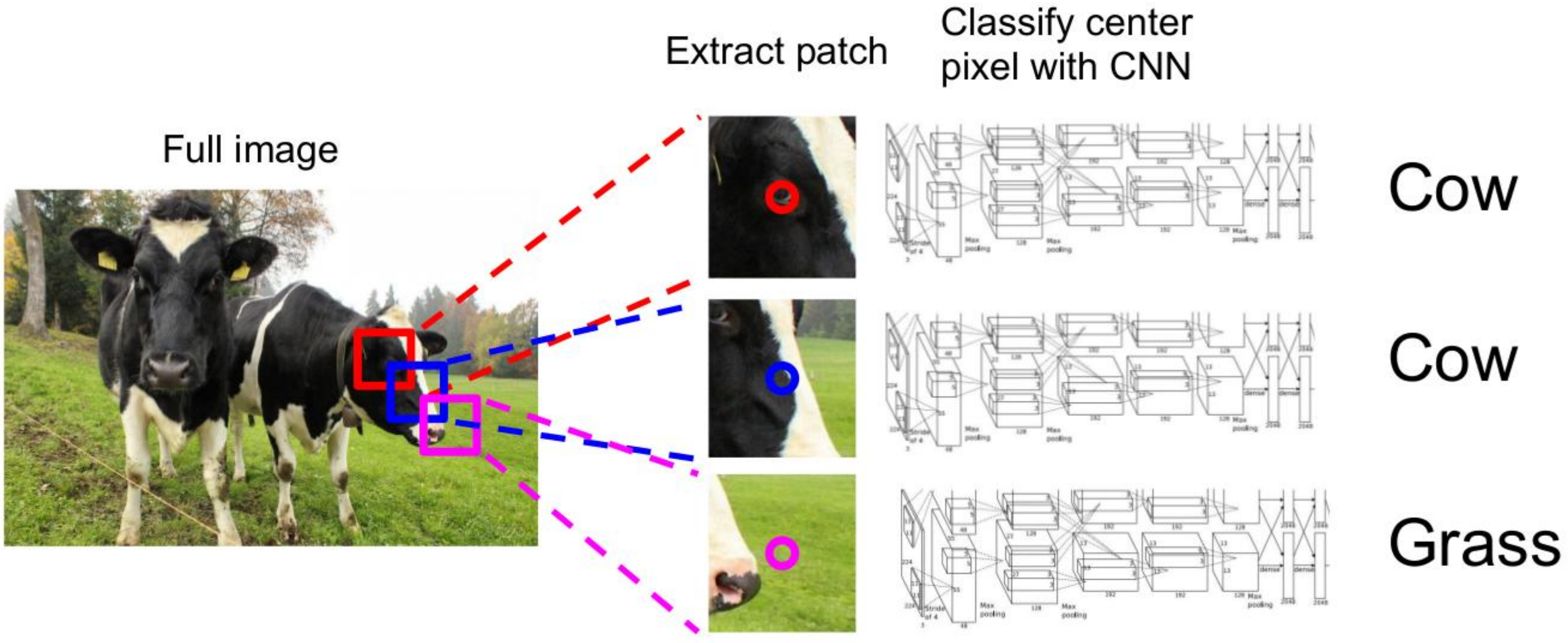
Techniques over time:

- Classical:
  - FCN (Fully Convolutional Network) – first to replace dense layers
  - U-Net – skip connections to recover spatial resolution
  - DeepLab series: atrous (dilated) convolutions and CRFs
- Modern: SegFormer, Detectron2, SAM2
- **Q?**

Fully Convolutional Network (FCN)

- Images have different sizes => Different size for the segmentation output
- How do you approach this?
  - Before: we used CNN + fully connected layers on top

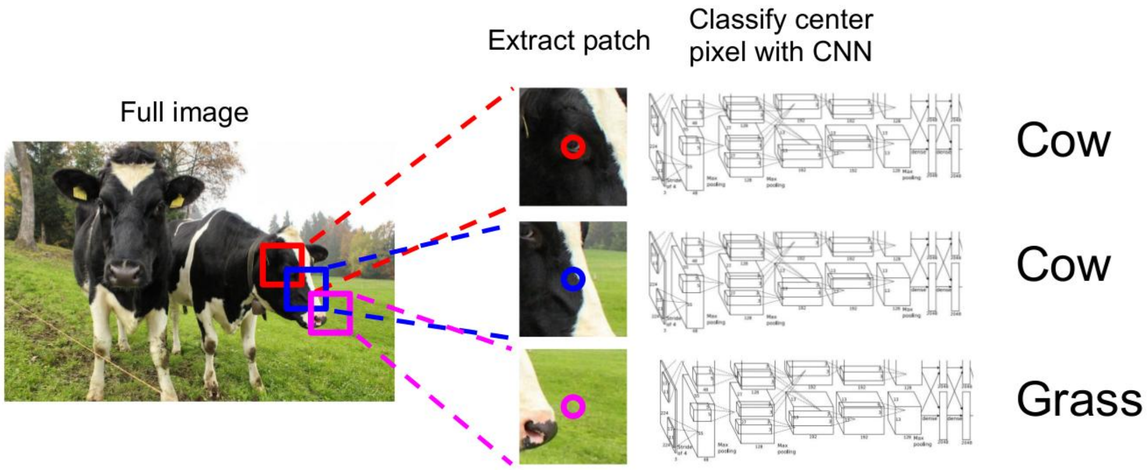
Segmentation: Sliding Window



Segmentation: Sliding Window

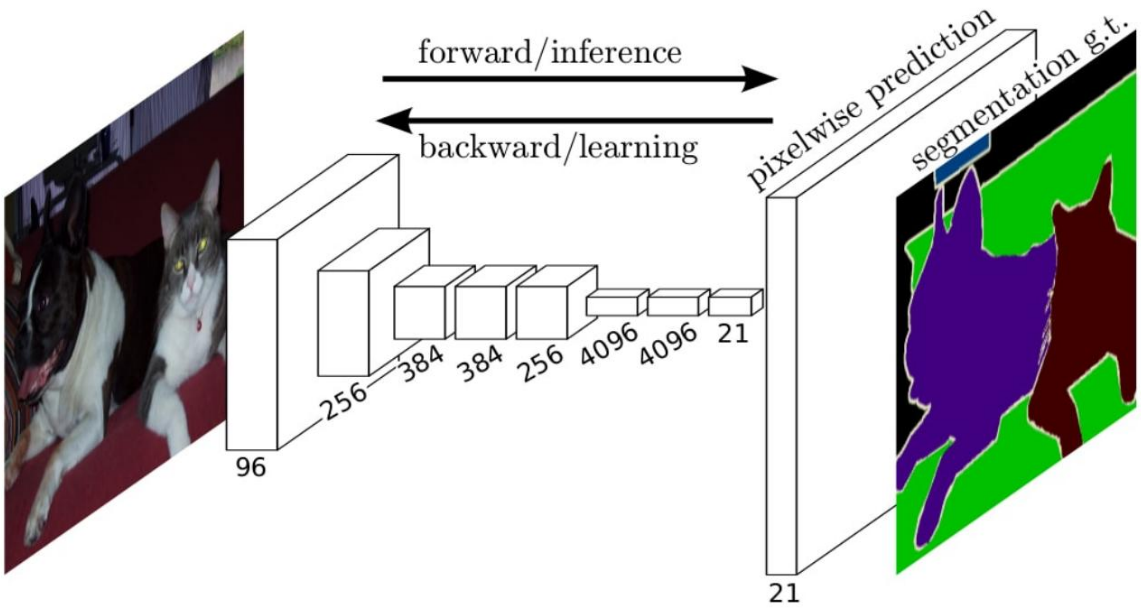
- Select patches obtained by sliding a window through the image
- Learn to classify each path as the class of the object or region found at its center
- Create (patches, center labels) pairs
  - Learn the convolutional network to predict the class of the center of a patch
- Classify **each pixel** by selecting a patch around it and run the learnt network

- Problem: Inefficient

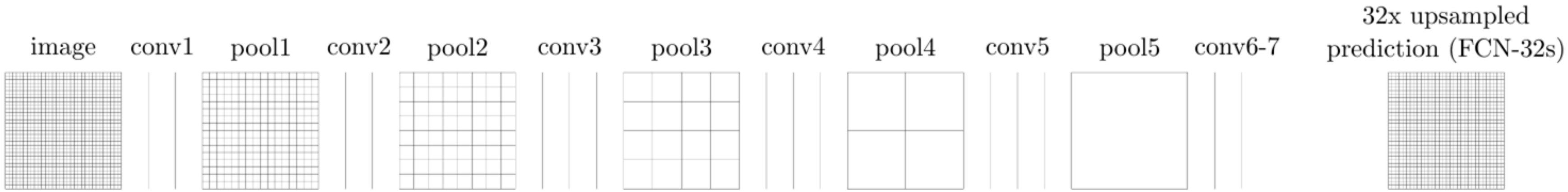


Fully Convolutional Network (FCN)

- All fully connected have been replaced by convolutions (they are equivalent)
- This network would work on any input size



Fully Convolutional Network (FCN)



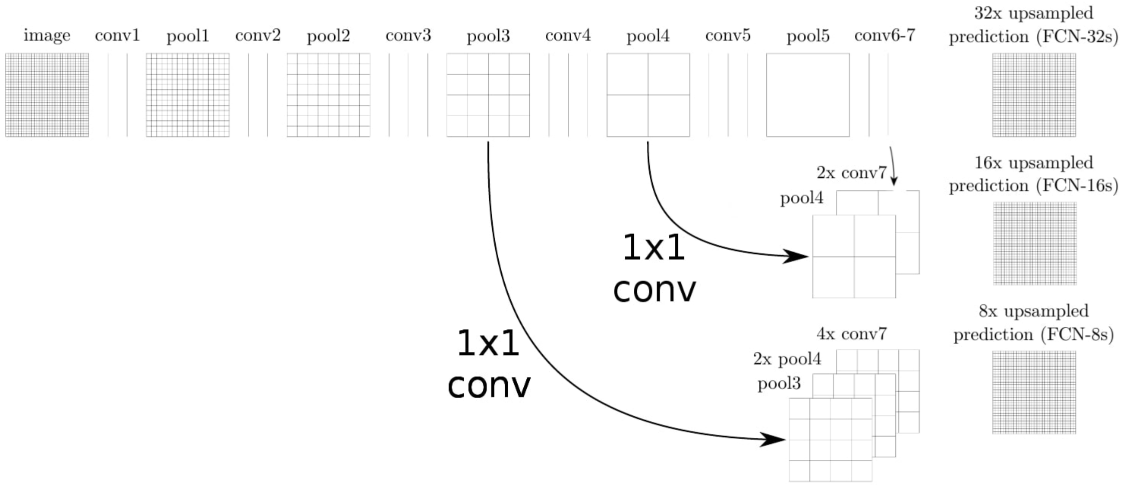
- The segmentation is rough because it results from upsampling *a low resolution* output

- *Solution*: use information from intermediate layers in the network, where low-level details are present

Long et al.[2014]: Convolutional networks for semantic segmentation.

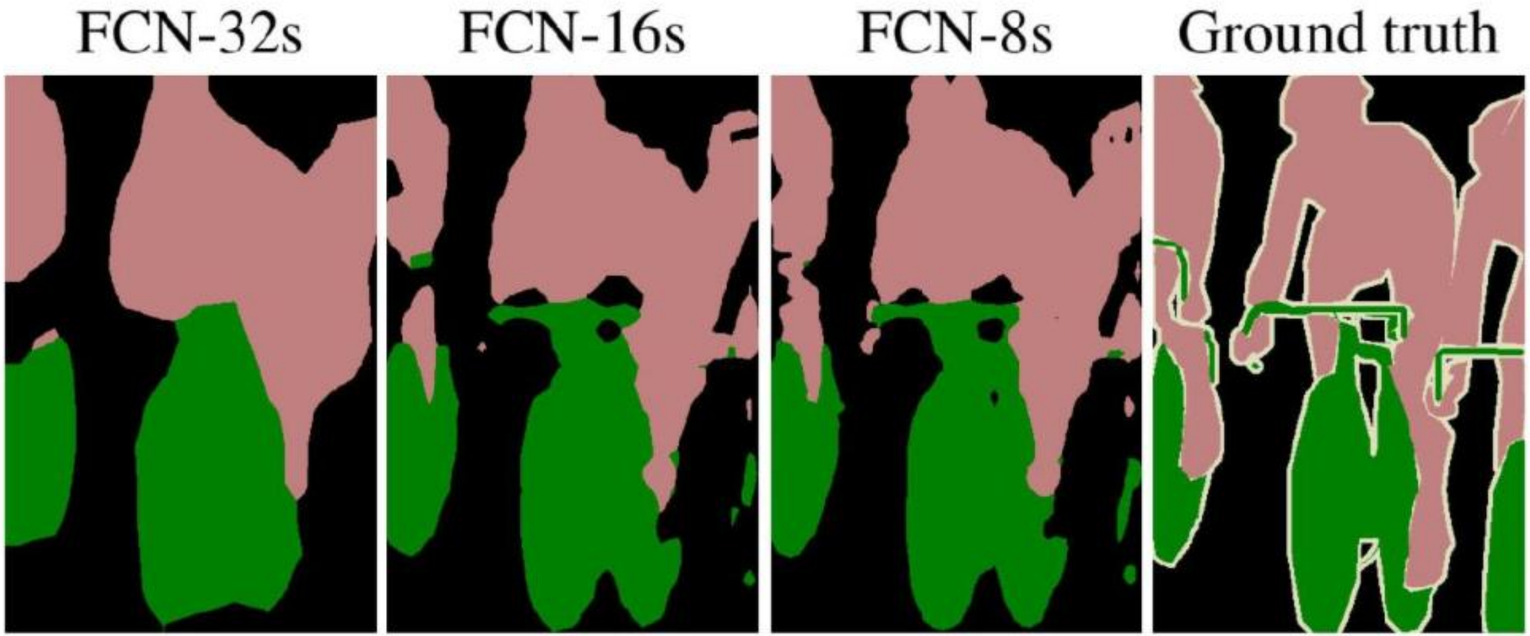
FCN: Skip Layers

- Combine the outputs of an upper and an intermediate layer
- Q: Why would low-level features help?
    - Useful for aligning the output with image boundaries, or with part of objects
- Use a 1x1 convolutional layer on features from lower levels to produce class predictions
    - Linear combination across channels
- Upsample and add prediction scores from different levels (then do a softmax)



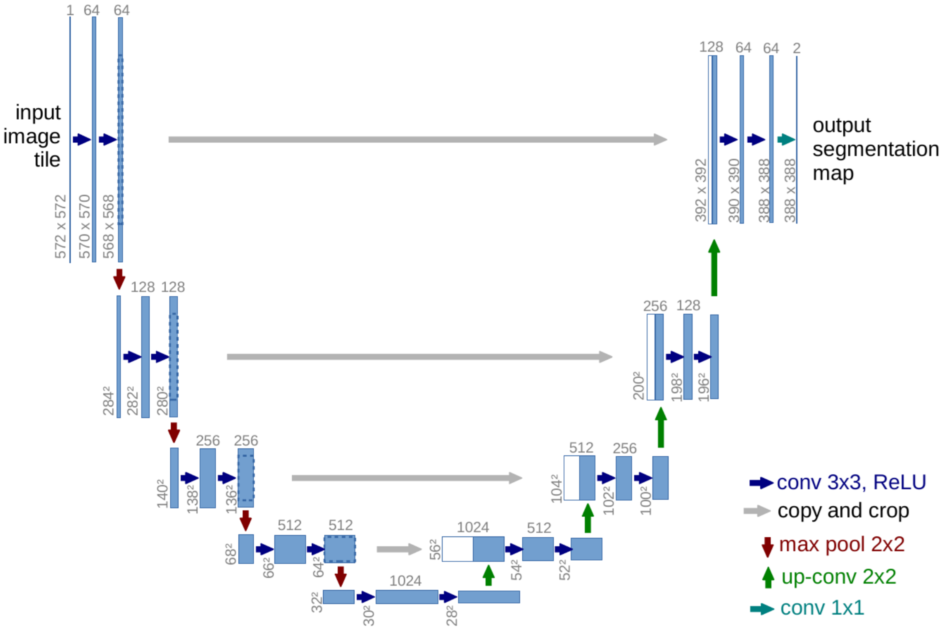
FCN: Results

- Using lower level features improves the level of details (resolution) of the output



U-net: using (again) intermediate features

- The contractive, convolutional part extracts information about the object
- Expansive part has the role of inferring the exac segmentation shape
- Upscale the convolutional maps in multiple layers
- The middle convolutional features are highly semantic but have low resolution - poor localisation
- Use previous convolutional features for fine-grained spatial information
- **Q: How is this different from FCN?**



Ronneberger [2015]: U-net: Convolutional networks for biomedical image segmentation

Instance segmentation

Task:

- Detect individual objects and segments each separately
- Each object instance is assigned a unique label

Techniques:

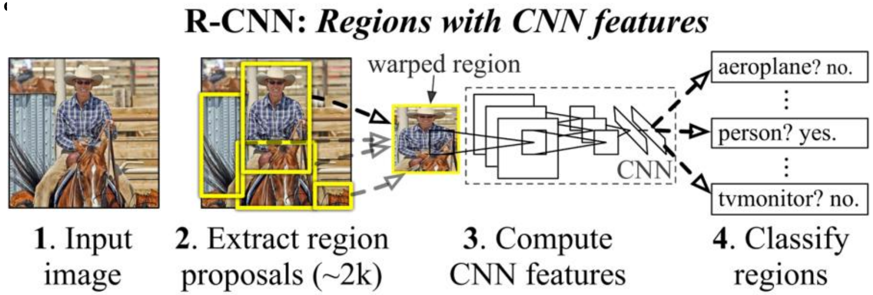
- Traditional: Sliding windows → R-CNN → Fast/Faster R-CNN
- Modern: Detectron2, RT-DETR, YOLOv12
- Usually **without** temporal information
  - **Video Instance Segmentation**: 3D CNNs, ConvLSTMs, MaskTrack R-CNN, TeViT

Supervised or Unsupervised

R-CNN, Fast R-CNN, Faster R-CNN, and Mask R-CNN

R-CNN (2014): Object Detection

- **Region-based Convolutional Neural Network**
- **Step 1**: Generate region proposals using **Selective Search** (not deep learning, pattern matching/descriptors)
- **Step 2**: Resize each proposal and extract features using a CNN (e.g., AlexNet).
- **Step 3**: Classify regions
- **Slow** due to processing each region independently.



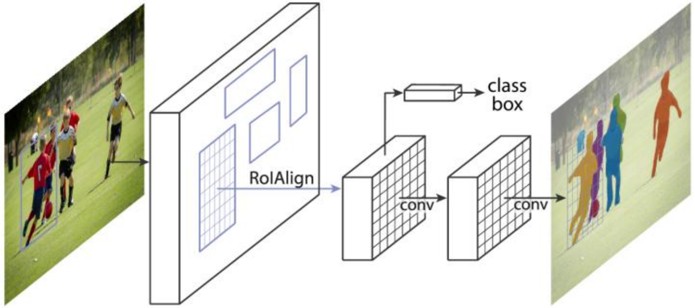
Fast R-CNN (2015): Object Detection

- **Single CNN pass**: Input image processed once through the CNN for feature extraction.
- RoI Pooling (Region of Interest) used to crop proposals directly from the feature map.
  - **Converts regions of varying sizes** into a **fixed-size output**.
- **End-to-end training** for both object classification and bounding box regression.
- **Faster** than R-CNN, but still not real-time.

R-CNN, Fast R-CNN, Faster R-CNN, and Mask R-CNN

Faster R-CNN (2015): Object Detection

- **Region Proposal Network (RPN)** to replace Selective Search (predicts anchor points + objectness score + bbox refinement)
- **End-to-end pipeline**: RPN generates proposals and object detection happens in a shared CNN.
- **Real-time object detection** with faster processing and better performance.
- Combines both **RPN** and **Fast R-CNN** for end-to-end object detection.



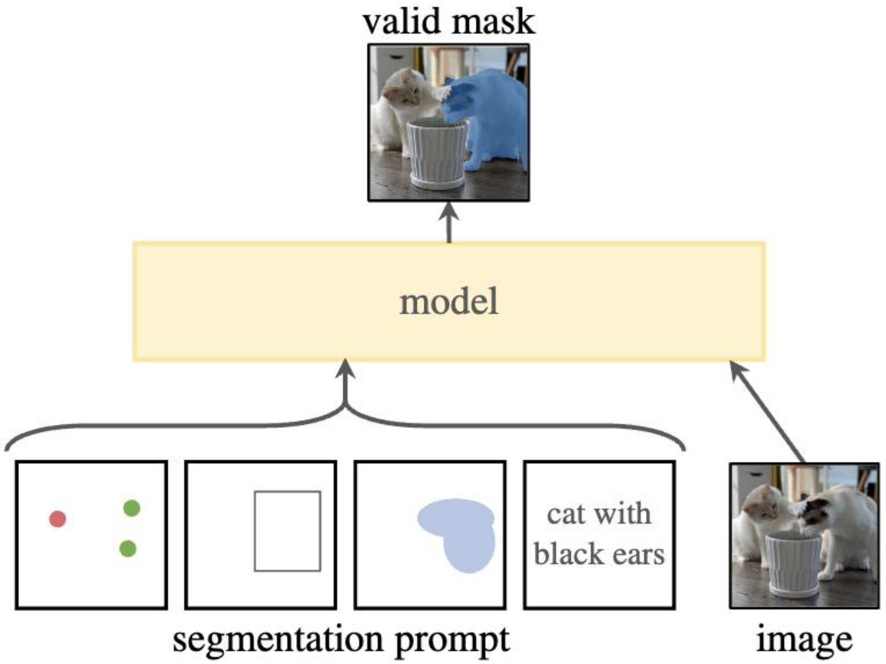
Mask R-CNN (2017): Instance Segmentation

- **Extension of Faster R-CNN** for **instance segmentation**
- Still uses **RPN**
- Adds a **branch for pixel-level segmentation masks** in addition to bounding box and classification.
- Uses **RoI Align** (instead of RoI Pooling) for better spatial accuracy.
  - Solves the **misalignment** issue present in **RoI Pooling**
- **Simultaneously** performs object detection and segmentation

Segment Anything Model (SAM)

Segment instances identified by:

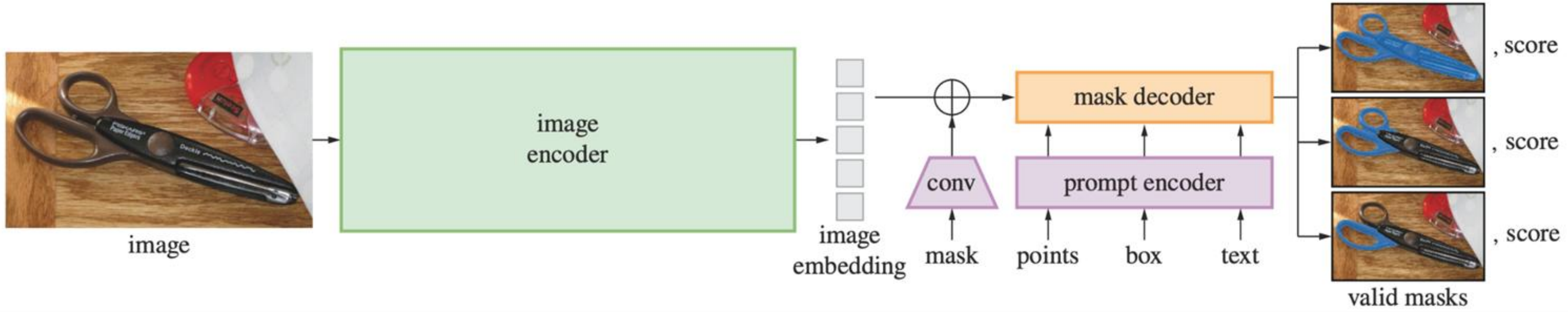
- Keypoints
- Boxes
- Rough maps (masks)
- Text prompts



Kirillov, et al. "Segment anything." *[2023]*

Segment Anything Model (SAM)

- In contrast, **new methods rely on large data** (rather than feature engineering and very complex pipelines) and **transformer based encoders and decoders**



Multi-Modal & Large Vision Models

Exploring the Future of Vision Systems

Large (Vision) Models

- **Parameters**: billions
- **Training data** size:
  - (e.g., ImageNet-21k: 14M, LAION 400M-5B, other proprietary data)
- **Model architecture**: typically are transformer-based
- **Training**: often in **self-supervised** or **multi-modal** settings
- **Purpose**: General visual understanding

Key Characteristics:

- **Scalability**: Capable of handling large datasets
- **Pre-training**: Leveraging pre-training techniques on vast datasets before fine-tuning on specific tasks
- **Generalization**: Can generalize across tasks with minimal task-specific adjustments

Shift in CV (based on those properties): From models that are **task-specific to foundation models**

Multi-Modal Learning

Task:

- models that can process and combine multiple types of data (e.g., text, images, audio).

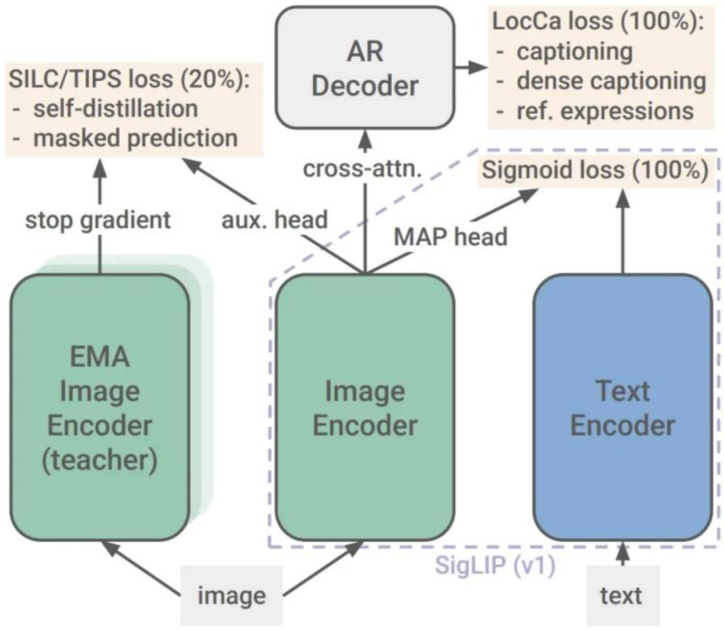
Technologies:

- Llama Vision, SigLIP2, PaliGemma, Gemini, Claude, DeepSeek-VL, ChatGPT
- Lots of open-weight foundation models

- **LLMs:** very **powerful**, very quick, just by predicting the next token, **WHY?**
- What can be improved by adding more modalities?
- Find concepts that **can not be understood** only from text!
- Could the consensus between the "senses"=modalities be the key, along with unsupervised learning?

SigLIP 2: Multilingual Vision-Language Encoders

- ViT architecture
  - learned positional embeddings
  - identical image and text encoders
- Losses1:
  - binary classification: image-text matching
  - image captioning
  - referring expression prediction (about a single object in a region)
  - grounded captioning (text for image regions)
- Losses2:
  - self-distillation (regions vs full image in teacher)
    - 8 students + 1 teacher
  - masked prediction in student
- Multiple resolutions
- Train on WebLI dataset (10B images + 12B alt-texts, 109 languages, 90% English)



Challenges in Multi-Modal Learning

- Modality Gap
  - Different modalities (e.g., images, text, audio) live in very different feature spaces.
- Data Imbalance & Noise
  - Some modalities (like images or text) are much more abundant than others (e.g., audio with aligned video); Noisy, mismatched pairs.
- Temporal Synchronization
  - Video + audio, or video + language, must be temporally aligned.
- Fusion Complexity
  - Choosing how and when to fuse modalities is still an open problem: Early fusion? Late fusion? Cross-modal attention?
- Missing Modalities
  - Real-world inputs often lack one modality
- Scalability
  - Large-scale multi-modal models (e.g., Flamingo, Gemini) require massive compute, memory, and multimodal data curation.
- Evaluation Benchmarks
  - Few standardized benchmarks exist that fully test cross-modal understanding
- Biases and Safety
  - Multimodal models inherit biases from all modalities.

Go to Viorica Patraucean & Razvan Pascanu lectures in NLP master after Easter.

Challenges & Future Directions

(in Computer Vision)

Challenges & Future Directions (my view)

Generalization/Robustness

- spurious correlations, OOD robustness, adversarial attacks

Data + Algorithm Efficiency

- Visual information is very reach (e.g. compare a book with a movie)
- Few-shot learning, self-supervised learning
- Model compression, quantization, other architectures (more video oriented maybe?)

Ethical issues

- bias, deepfakes, hallucination in vision-language models => Explainability and Trust

Multi-modal - for sure (see previous slides)

Privacy? Federated & privacy-preserving vision

Reminder to talk about: Internships 3-4 months **ML research -->** part remains in research, part in engineering (in other Bitdefender teams)

Thank you!
(Next: NLP Applications & Generative Models)