Neural Network Architectures

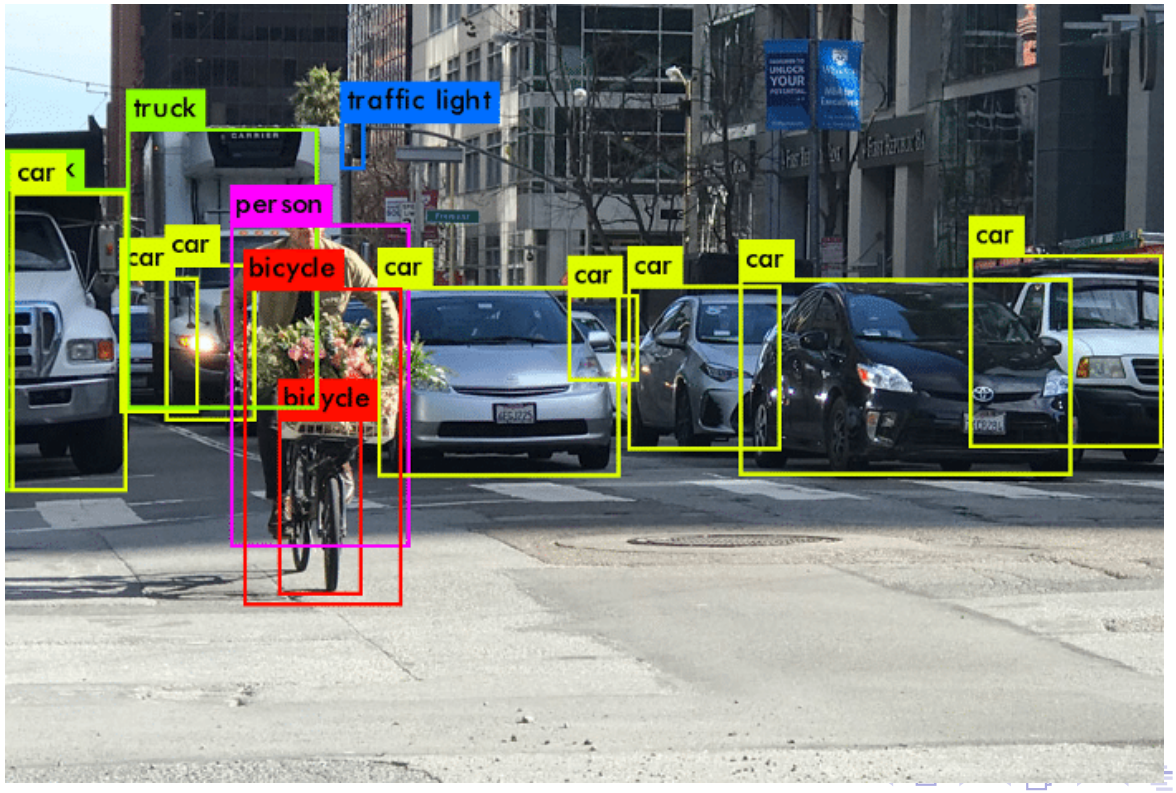
Beurton-Aimar

October 14, 2025

Finding Objects

Next step after segmentation

- In a lot of application, segmentation is only done to extract object.
- A new approach prooses to find objects inside an image/video and to return the bounding box of each.



Yolo - The Story

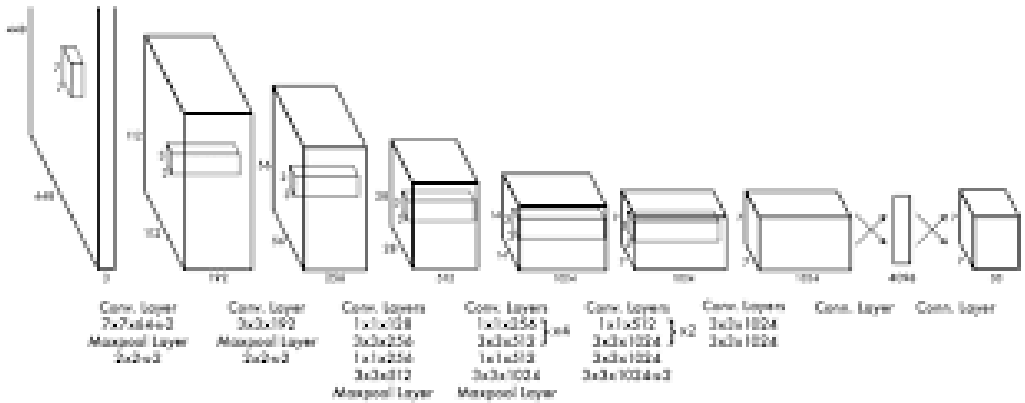
First version

- Yolo - You Only Look Once - has been introduced in 2015 by Joseph Redmond et al (https://arxiv.org/pdf/1506.02640) at the CVPR conference.
- Before Yolo, the most popular models based on R-CNN need to analyze several times the image to extract all features.
- The main idea is to see only one time each pixel belonging to the image in order to predict the position of each object.
- The gain of performances has allowed to use the technology to analyze video in real time. The first version has been able to treat 115 images per second.

Yolo - The Story

From version 1 to version 3

- The first version is made of:
  - 24 convolutional layers with Maxpool layer between them.
  - 2 fully connected layers to produce outputs.
- YoloV2 proposed some improvements:
  - Batch normalisation
  - Able to get largest images as input.
  - Anchor boxes to detect objects with different sizes.
- YoloV3 has changed its backbone to use Darknet-53.



Yolo - The Story

What is a backbone?

- A backbone is the part of the network in charge of extracting visual features from images.
- Could be considered as the **eye** of the model.
- It converts the raw image in usefull information to track and to identify objects.

YoloV3

- A new possibility to detect objects at multi-scales. That improves to recognition of very small or very large objects in the same image.
- YoloV3 also improved the loss function to correct localization and classifiction errors.
- It was the last work of J. Redmon who decided to stop research in this domain because of the possible usage of such tools to stack people (military usage and so on).

Yolo - The Story

A new teaam - Ultralytics

- Alexey Bochkovskiy with Ultralytics has produced the version V4 (2020).
- This version is coded in PyTorch and stays open source.
- Open the new age in the Yolo history.
- Until 2022 several versions of Yolo are published but the V7 mark a real new age in july 2022 with a new architecture.
- In 2023, publication of YoloV8 with a big success.

Yolo - Howto

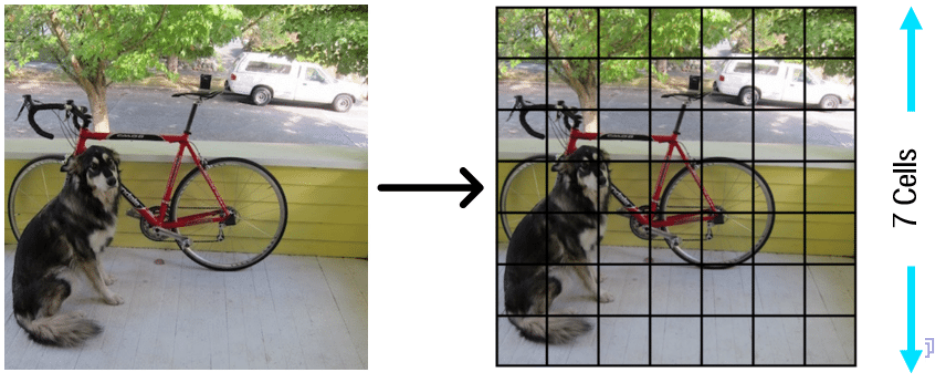
Design and Functionning

- 5 main concepts are implemented in Yolo:
  1. Applying a grid to cut out the image.
  2. Using a backbone.
  3. Creating anchor boxes.
  4. New output format.
  5. Post-treatment using Non-Max suppression algorithm (NMS)

Yolo - Howto

Slicing with a grid

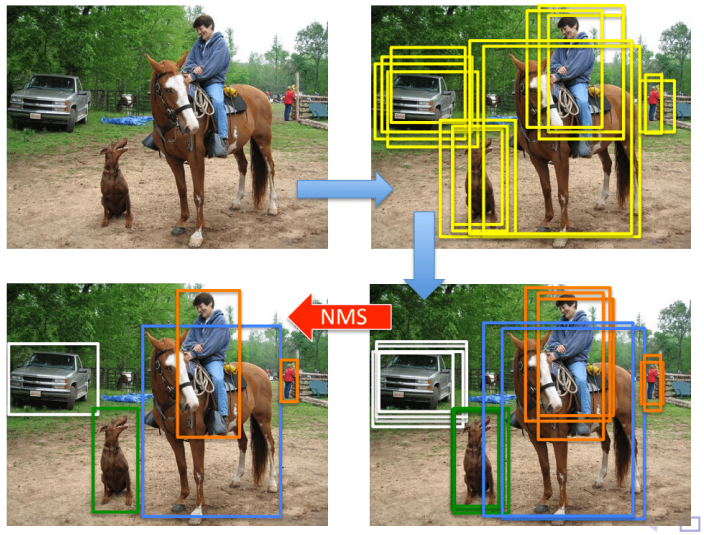
- Yolo slices the input image in cells grid.
- Each cell has to detect object which the center is in this area.
- In the cell, the model predicts one or several bounding boxes defined by their positions and their size (width and height).
- For each box, a score is set according the confidence for the presence of the object inside the box and the alignment with the real object (ground truth).
- The cell predicts the object label.



Yolo - Howto

Backbone

- Based on ResNet, with residual blocks.
- From YoloV12, it contains an attention module, ELAN module at the beginning.
- In the last version, the attention module is a local attention, R-ELAN module.
- The last step, NMS, allows to only keep the “best” boxes.



Attention Mechanism

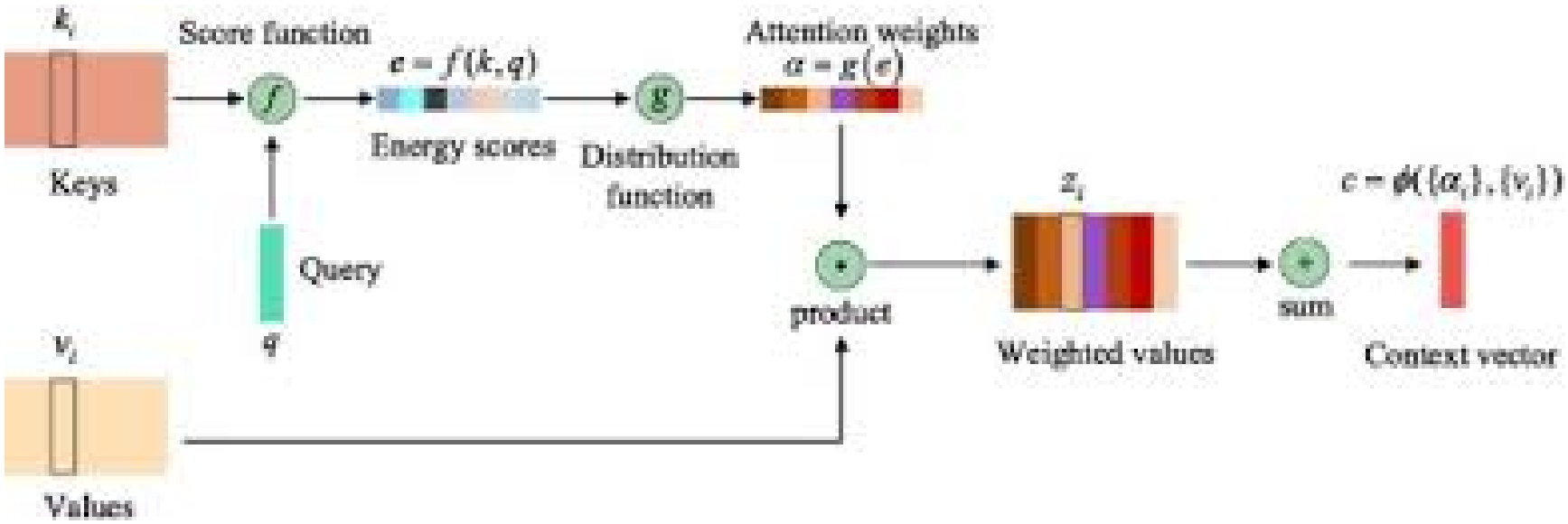
Story

- First article: "Attention is All you need" (Vaswani et al. 2017).
- The next stage after recurrent network architecture.
- Introduce transformer networks, main usage for LLMs models.

Description

- The main principle: finding which parts of the input is the main one and telling to the model to focus on it, and ignoring the other one.
- 3 main concepts: query, key and value.
- **Query vector** contains information you are searching for.
- **Key vector** contains information from the token and be used to compute the weights of each attention.
- **Value vector** weights key vector depending on the alignment between request and key.

Attention Model



from "A review on the attention mechanism of deep learning", Niu et al, Neurocomputing (2021)

Go Back to Yolo

Feature	Purpose	Benefit
Area Attention	Local attention within regions	Fast yet context-aware attention
R-ELAN	Residual architecture with better stability	Trains deep models reliably
Flash Attention	Optimized memory-efficient attention	Lower latency, faster inference
Conv-first design	CNN-friendly transformer hybrid	Retains speed + GPU efficiency
No pos encoding	Lightweight position awareness	Simpler and faster architecture

Recap: YoloV12's key innovations

from https://pyimagesearch.com/2025/07/07/breaking-the-cnn-mold-yolov12-brings-attention-to-real-time-object-detection/

Transfer Learning

Not enough data!

- Most of the time, neural networks need a lot of data.
- Data are expansive, difficult to collect.
- Data augmentation are not the only way to solve the problem.
- Using pre-trained model is usual: transfer learning is the way to do.

Transfer learning vs Fine-Tuning

- Other way to do by fine-tuning.
- Fine-tuning allows some or all of the pre-trained model's layer to be retrained on a new dataset.

Transfer Learning

Description

- Only final layers are retrained - the rest of the model is frozen.
- Low computational cost due to only final layers are trained.
- Limited adaptation to new tasks.
- Lower risk of overfitting with smaller datasets.

Usage

- Small dataset.
- The new task closely resembles the original task for example classifying different types of images.
- A quick solution with limited computational resources is needed.

Fine Tuning

Description

- Allowing some or all pre-trained model's layers to be retrained (adjusted) on a new dataset.
- Help the model better adapt to the specifics of the new task.
- Require more data and computation than transfer learning.

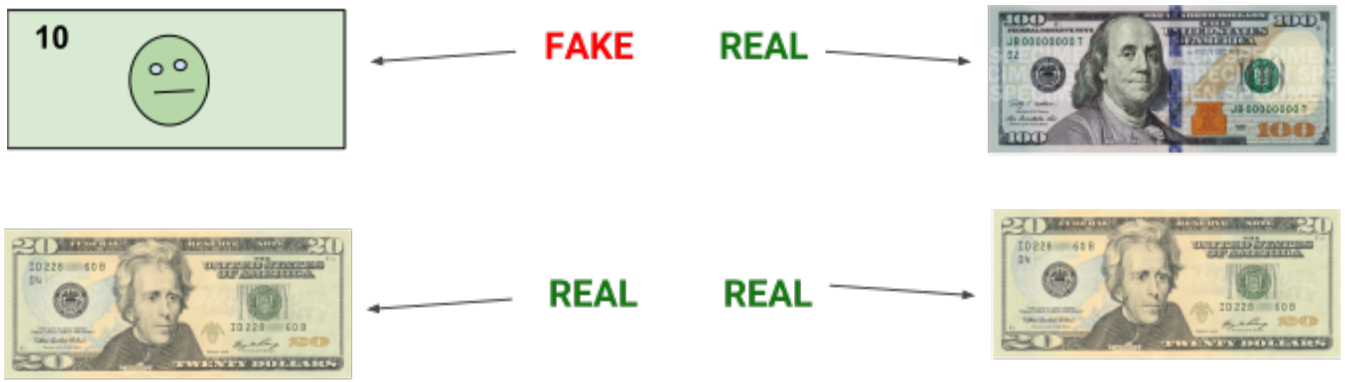
Usage

- Entire model or specific layers are retrained allowing more adaptation.
- Require more data than transfer learning.
- More computationally expensive due to retraining more or less the entire model.
- High risk of overfitting with small datasets.
- Using fine tuning when the new task differs significantly from the original and requires deep model adaptation.

GAN - Generative Adversarial Networks

Definition

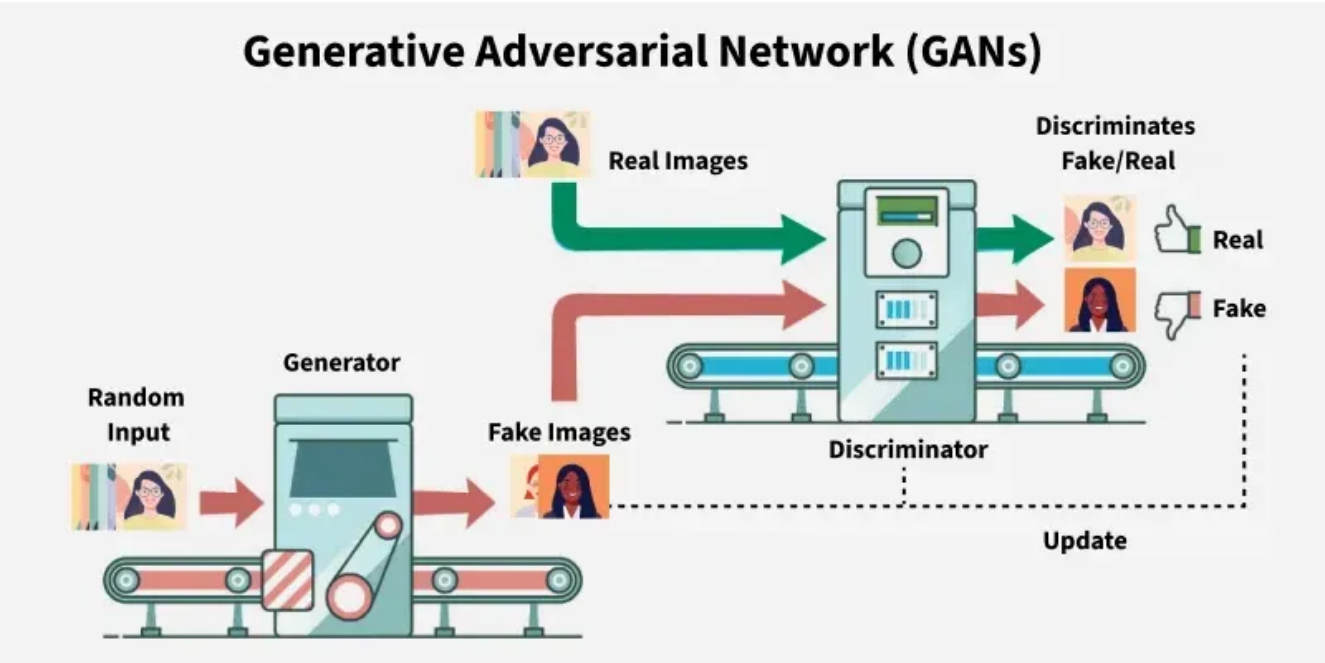
- A generative adversarial network (GAN) has two parts:
  - The **generator** learns to generate plausible data. The generated instances become negative training examples for the discriminator.
  - The **discriminator** learns to distinguish the generator's fake data from real data. The discriminator penalizes the generator for producing implausible results.



GAN

Description

- Both generator and discriminator are neural networks.
- Generator output is connected directly to the discriminator input.
- Through the backpropagation the discriminator's classification provides a signal that the generator uses to update its weights.



GAN

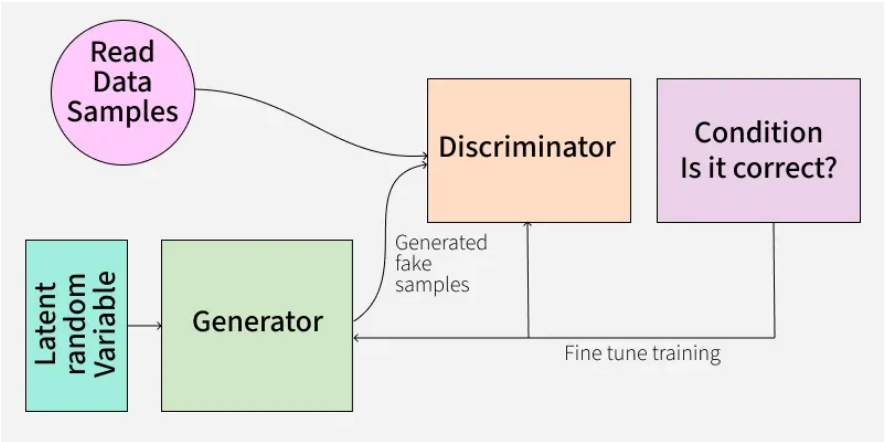
How it works

- The generator starts with a random noise vector like random numbers. It uses this noise as a starting point to create a fake data sample such as a generated image.
- The discriminator receives two types of data:
  - Real samples from the actual training dataset.
  - Fake samples created by the generator.
- Adversarial Learning:
  - If the discriminator correctly classifies real and fake data it gets better at its job.
  - If the generator fools the discriminator by creating realistic fake data, it receives a positive update and the discriminator is penalized for making a wrong decision.

GAN

How it works

- Generator's improvement:
  - Each time the discriminator mistakes fake data for real, the generator learns from his success.
  - Through many iterations, the generator improves and creates more convincing fake examples.
- Discriminator's adaptation:
  - The discriminator learns continuously by updating itself to better spot fake data.



GAN

Several Models

- VanillaGan is the base model - both generator and discriminator are multi-layers perceptron (MLPs).
- DCGAN to produce new images.
  - Uses Convolutional Neural Networks (CNNs) instead of simple multi-layer perceptrons (MLPs).
  - Max pooling layers are replaced with convolutional stride.
  - Fully connected layers are removed to obtain better spatial understanding of images.
- BigGANs are used to produce high quality video.

GAN

Advantages

- Synthetic Data Generation: GANs produce new, synthetic data resembling real data distributions which is useful for augmentation, anomaly detection and creative tasks.
- High-Quality Results: They can generate photorealistic images, videos, music and other media with high quality.
- Unsupervised Learning: They don't require labeled data helps in making them effective in scenarios where labeling is expensive or difficult.
- Versatility: They can be applied across many tasks including image synthesis, text-to-image generation, style transfer, anomaly detection and more.