MO434 - Deep Learning
Applications in Image Analysis

Alexandre Xavier Falcão

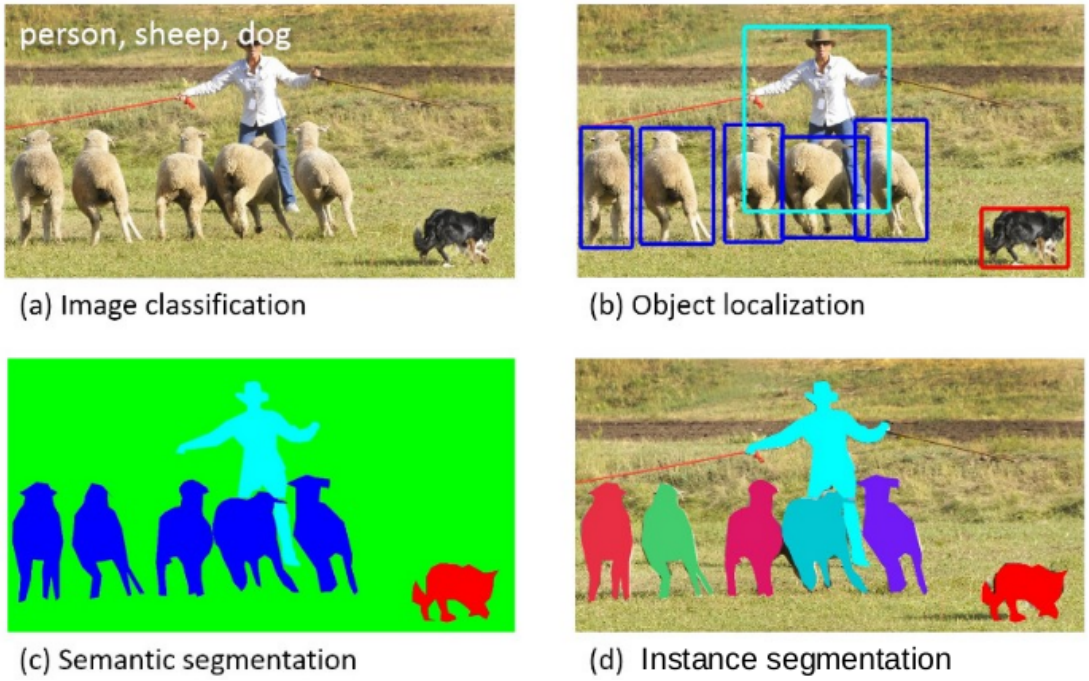
Institute of Computing - UNICAMP

afalcao@ic.unicamp.br

Introduction

CNNs have several applications in image analysis, but they are mostly based on:

- image classification,
- object detection (localization),
- semantic and instance segmentation.



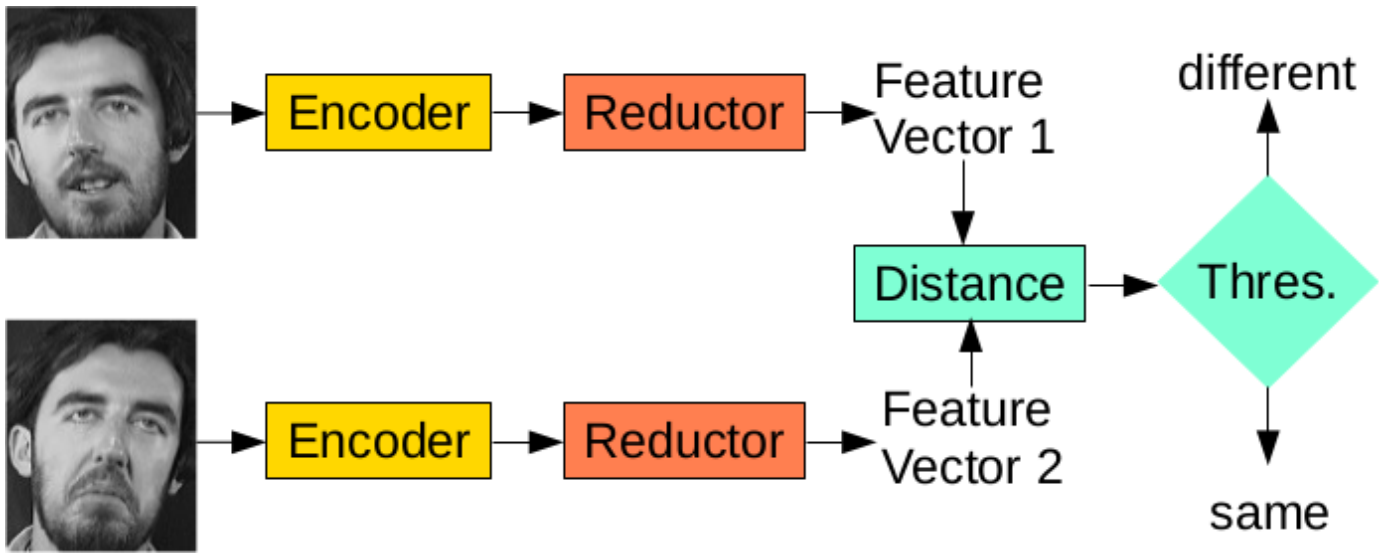
(a) Image classification (b) Object localization

(c) Semantic segmentation (d) Instance segmentation

Figure extracted from [1].

Introduction

- We have used CNNs to build predictive models for image classification.
- In some applications (e.g., biometrics), we may want to compare visual representations and verify whether or not they belong to a same class.



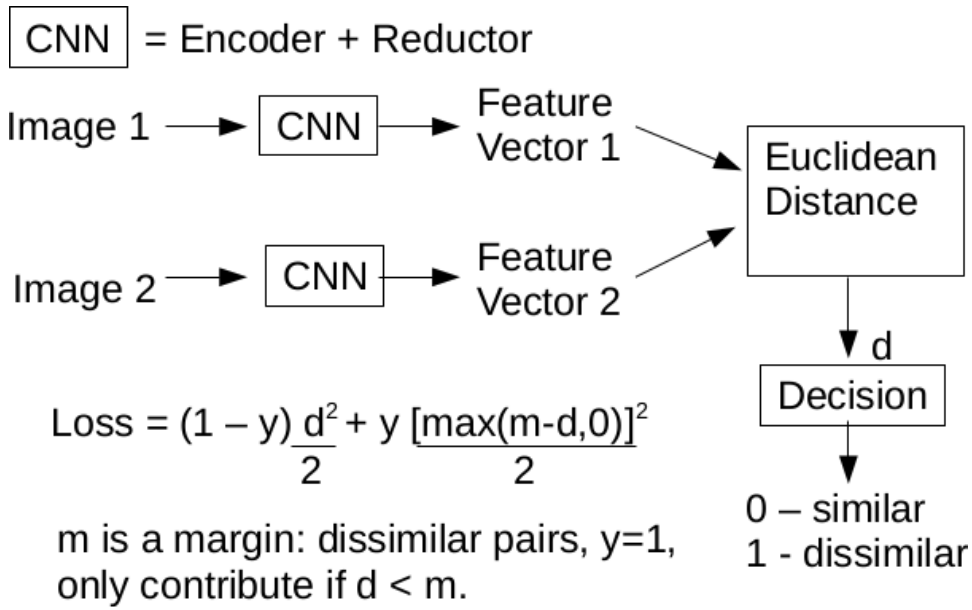
Such models are called contrastive, a Siamese network [2].

Agenda

This lecture will cover

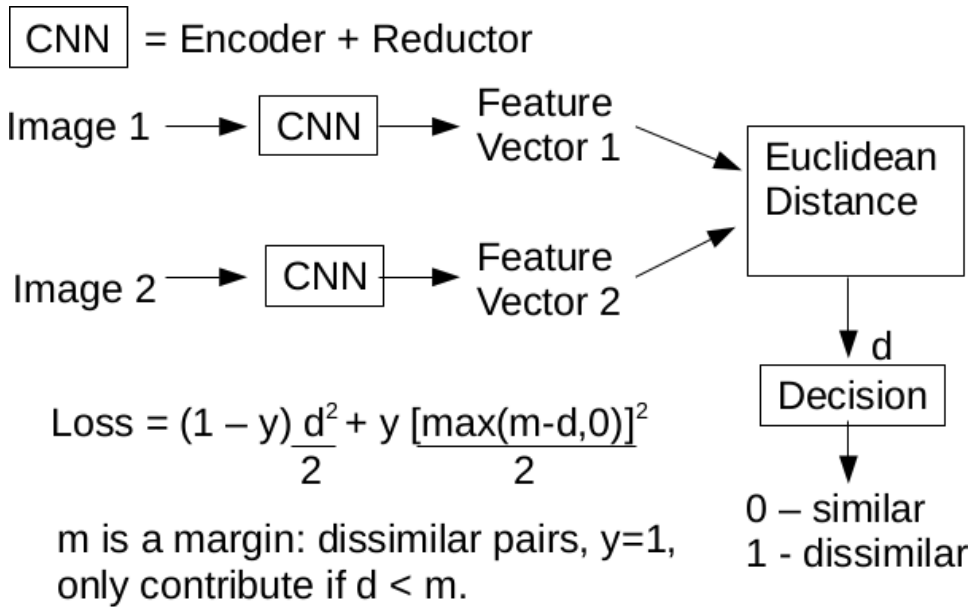
- a contrastive model for face verification,
- the evolution of an efficient model for object detection, and
- the most popular network's shape for semantic segmentation.

Building a contrastive model



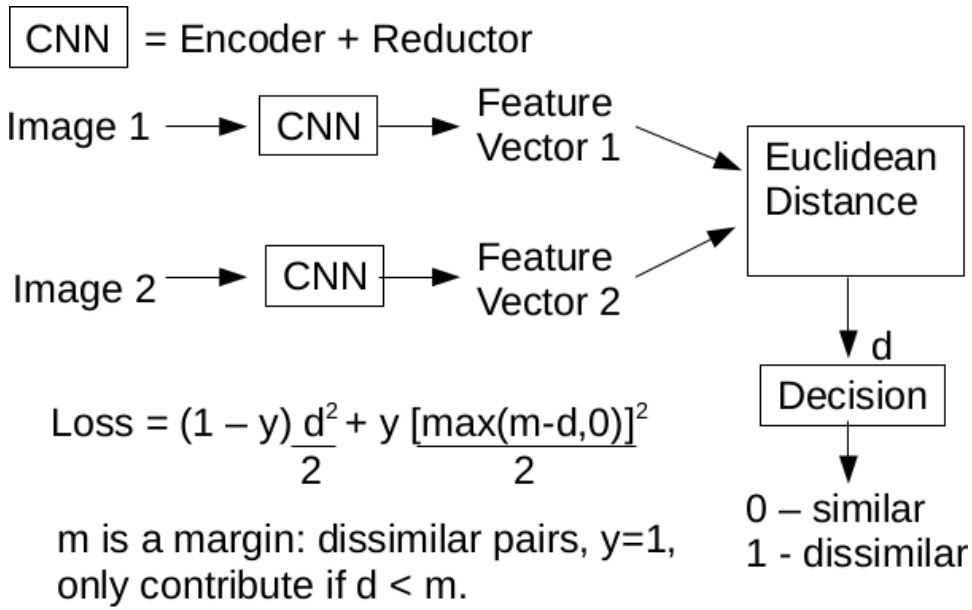
- The encoder may be pre-trained or trained from scratch.

Building a contrastive model



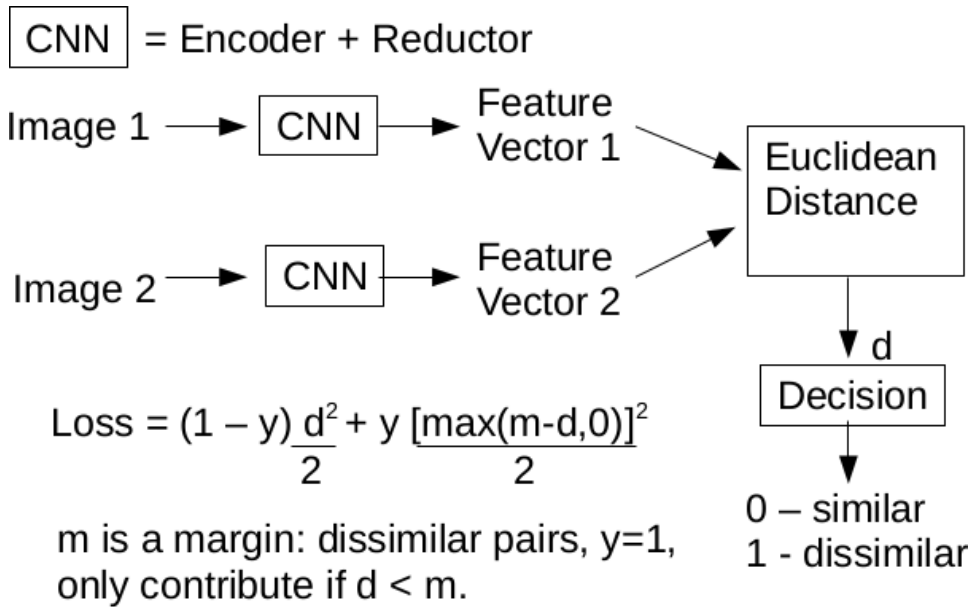
- The encoder may be pre-trained or trained from scratch.
- The reductor uses dense layers to reduce dimensionality.

Building a contrastive model



- The encoder may be pre-trained or trained from scratch.
- The reductor uses dense layers to reduce dimensionality.
- There are better distance and loss functions [3, 4].

Building a contrastive model



- The encoder may be pre-trained or trained from scratch.
- The reductor uses dense layers to reduce dimensionality.
- There are better distance and loss functions [3, 4].

Let's build a Siamese network for face verification.

▸ (CONTRASTIVE LEARNING).

Object detection

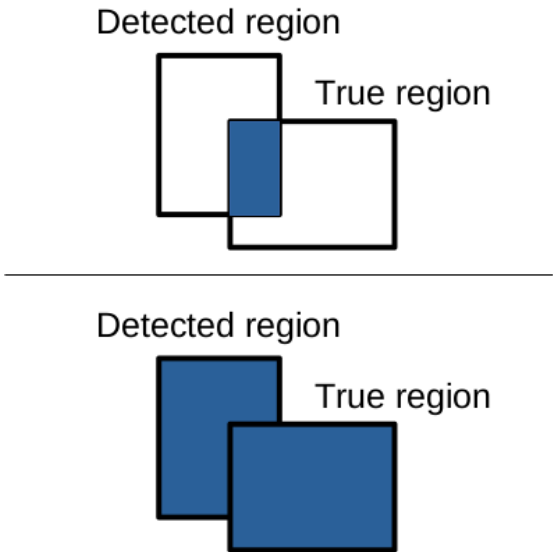
Object detection models usually regress a bounding box around each object location and classify the object inside.

from viso.ai

A review of object detection with deep learning can be found in [5] and codes can be found in `https://paperswithcode.com/`.

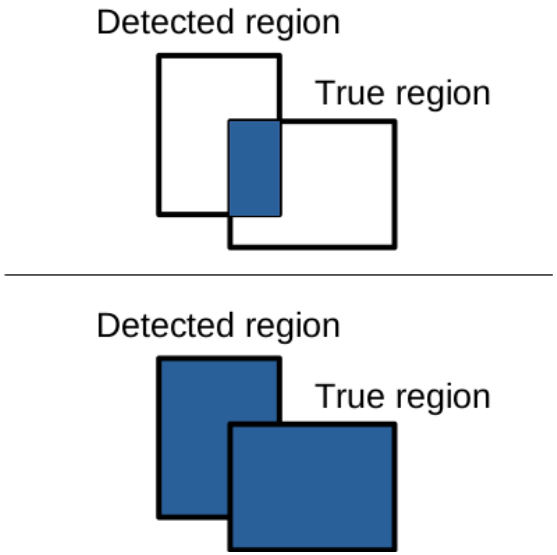
Measuring effectiveness in object detection

- Success can be measured by Intersection over Union (IoU).



Measuring effectiveness in object detection

- Success can be measured by Intersection over Union (IoU).

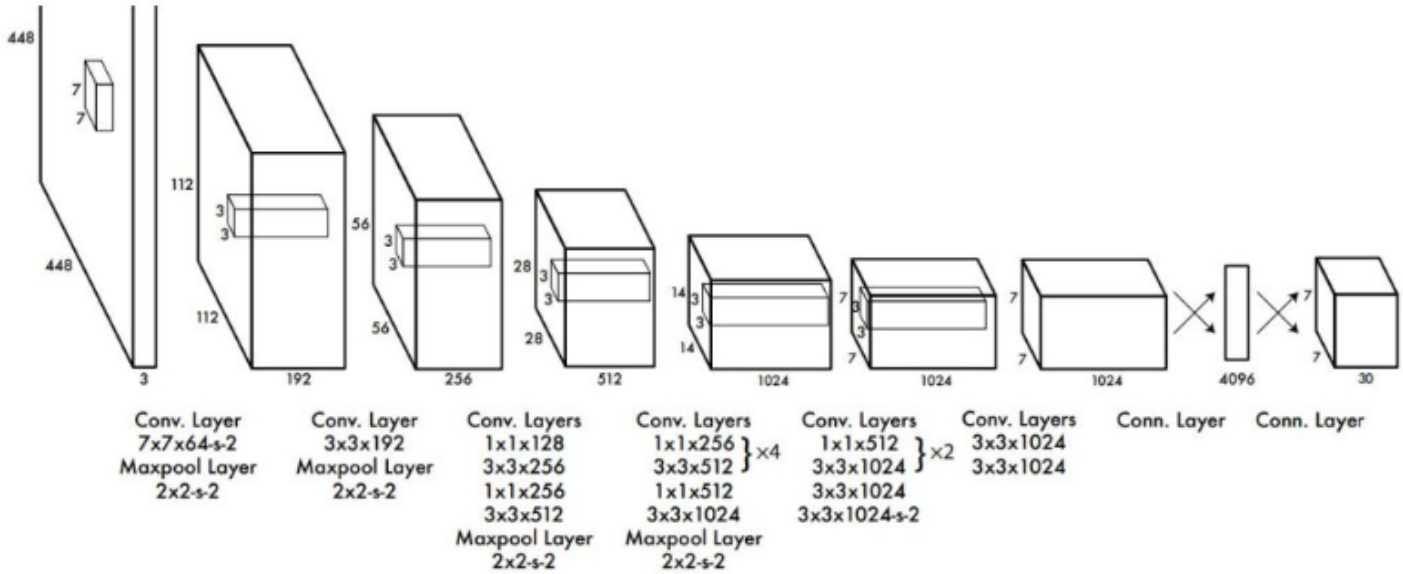


- Precision is the number of true positives (bounding boxes that led to correct prediction) divided by the sum of true positives and false negatives.

- For various IoU thresholds, one can measure average precision (AP) for each class and the mean of AP across classes is the effectiveness measure called mean average precision (mAP).

A fast model for object detection

YOLO (You Only Look Once) is a family of real-time object detection models from v1 (2015), based on CNN, to v11 (2024) that uses CNN and visual transformer (see yolo-explained in viso.ai).



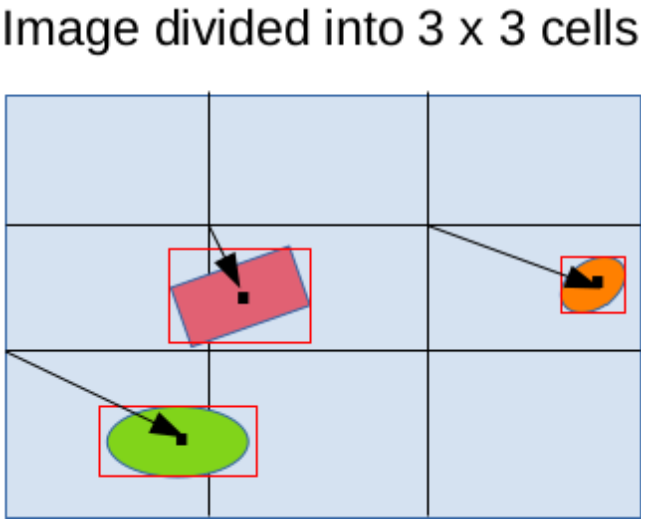
Original v1 from `https://arxiv.org/pdf/1506.02640`.

An encoder with 22 layers extracts features while 2 fully connected layers predict the output class probabilities and coordinates.

A fast model for object detection

Assuming one object per cell, in YOLOv1, we

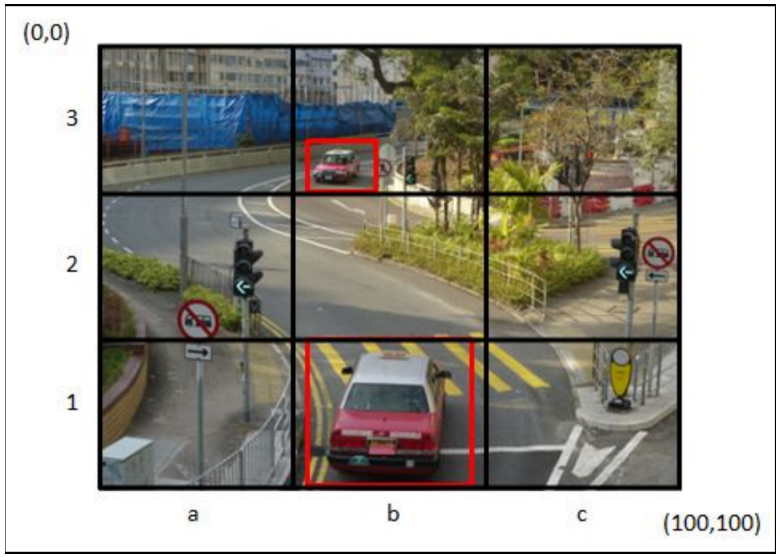
- divide each image into $N \times N$ cells,
- identify which cells contain the center of a ground-truth bounding box, and
- train a CNN to output $N \times N$ estimates of class, proportional size and relative offset of the objects in an image.



A fast model for object detection

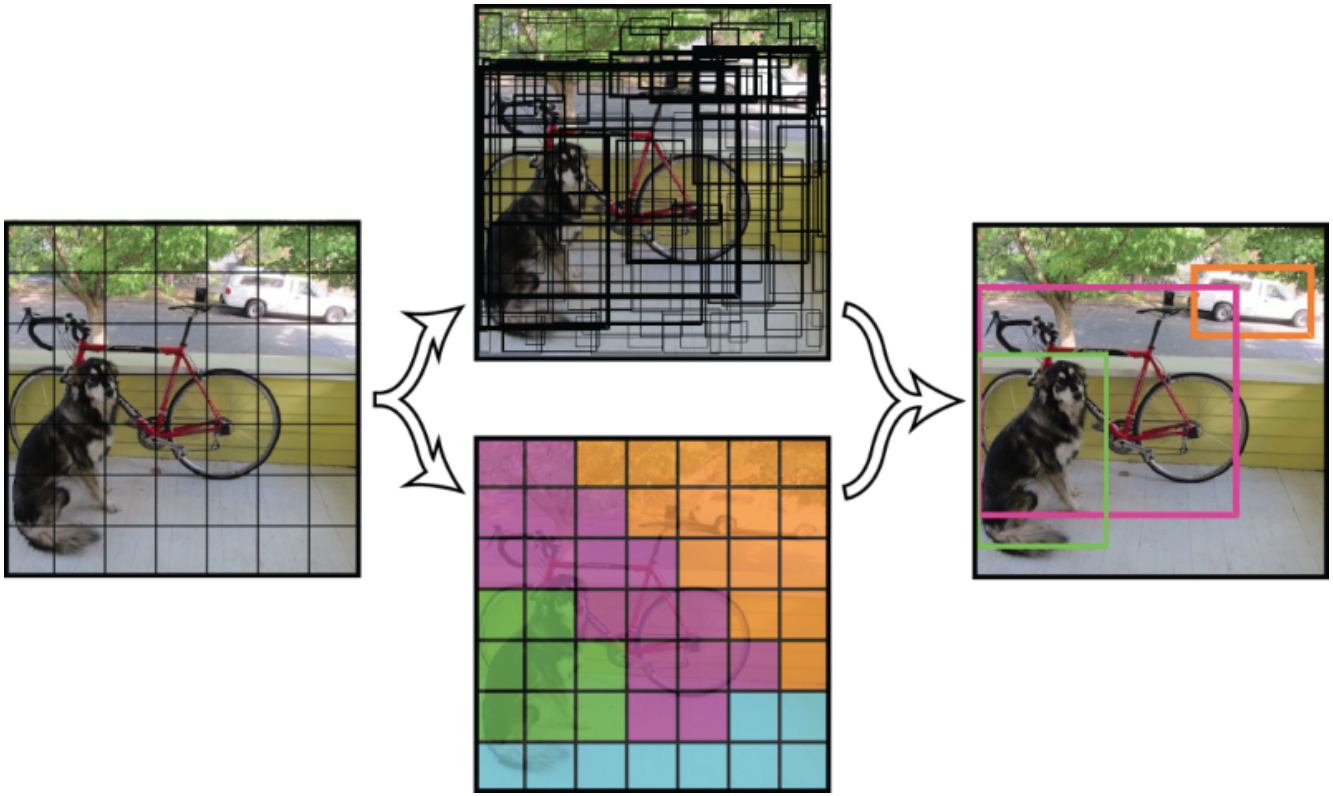
Assuming one object per cell, in YOLOv1, we

- divide each image into $N \times N$ cells,
- identify which cells contain the center of a ground-truth bounding box, and
- train a CNN to output $N \times N$ estimates of class, proportional size and relative offset of the objects in an image.



A fast model for object detection

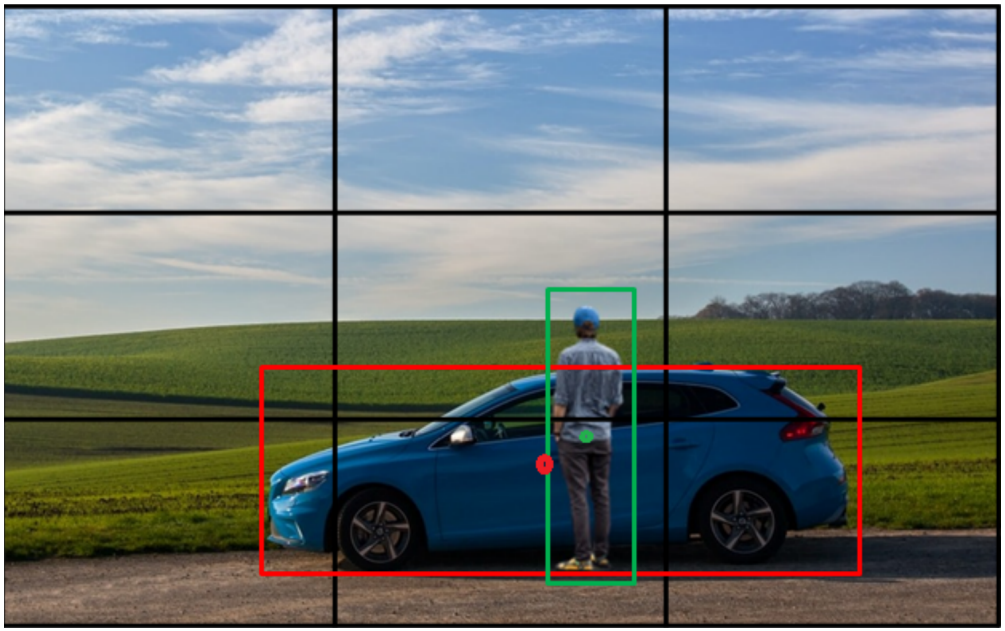
YOLOv1 estimates a class probability (below) and bounding box (above) from each cell (left) and selects the highest-probability bounding box from each class (right) as result.



from `https://arxiv.org/pdf/1506.02640.`

A fast model for object detection

Subsequent versions can deal with multiple object centers per cell by estimating bounding boxes of different aspect ratios to represent them (anchor boxes).



The YOLO family of models can be more easily created with `https://github.com/ultralytics/ultralytics`.

▸ (FINE-TUNNING YOLOv11).

Fully Convolutional Neural Networks (FCNN)

- So far we have learned that convolutional layers (backbone, encoder) extract suitable image features while predictions (classification or regression) are done by dense layers.

Fully Convolutional Neural Networks (FCNN)

- So far we have learned that convolutional layers (backbone, encoder) extract suitable image features while predictions (classification or regression) are done by dense layers.

- The strategy is suitable for image classification and object detection. For segmentation, however, we have to classify pixels as belonging or not to each object of interest.

Fully Convolutional Neural Networks (FCNN)

- So far we have learned that convolutional layers (backbone, encoder) extract suitable image features while predictions (classification or regression) are done by dense layers.

- The strategy is suitable for image classification and object detection. For segmentation, however, we have to classify pixels as belonging or not to each object of interest.

- You may wonder to train one classifier (dense layers) per pixel by using its values in the feature map, but well succeeded approaches substitute such predictors per pixel by a decoder.

Fully Convolutional Neural Networks (FCNN)

- So far we have learned that convolutional layers (backbone, encoder) extract suitable image features while predictions (classification or regression) are done by dense layers.

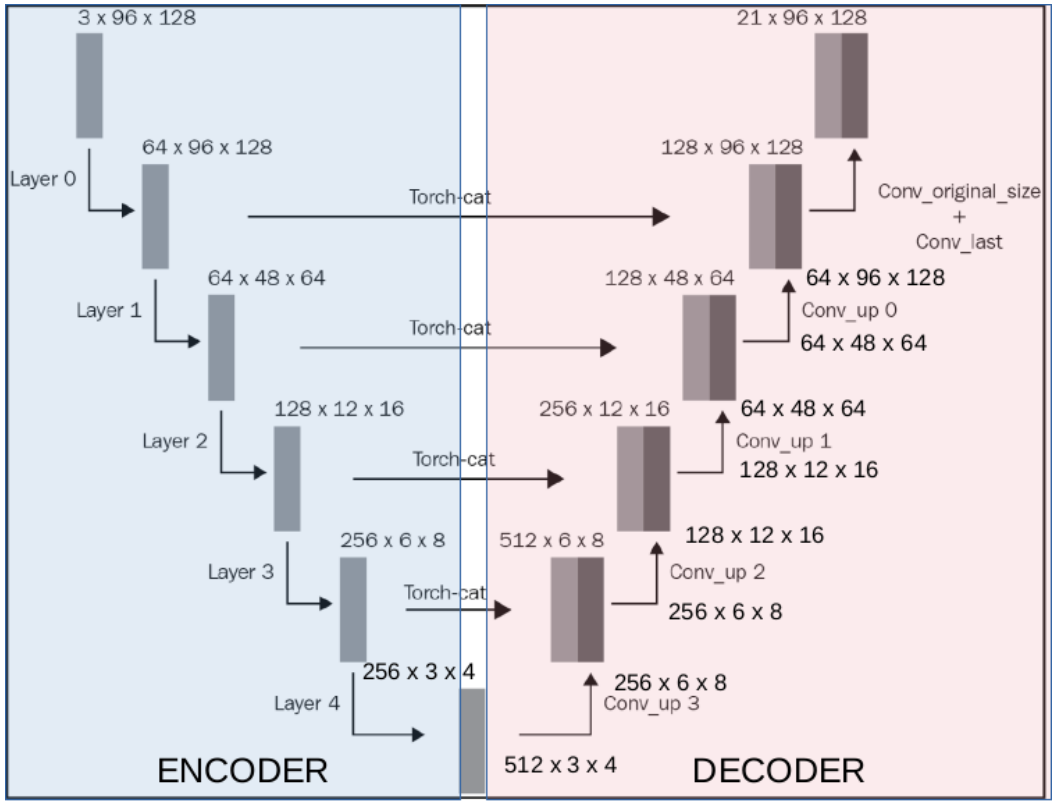
- The strategy is suitable for image classification and object detection. For segmentation, however, we have to classify pixels as belonging or not to each object of interest.

- You may wonder to train one classifier (dense layers) per pixel by using its values in the feature map, but well succeeded approaches substitute such predictors per pixel by a decoder.

- While the encoder reduces image size, the decoder must retrieve the original spatial dimension with one output channel per class.

Fully Convolutional Neural Networks (FCNN)

A decoder is also a sequence of convolutional layers, except that up-sampling is adopted to retrieve spatial dimension.



U-Net is one of the most well succeeded and popular FCNNs (`https://paperswithcode.com/method/u-net`).

Semantic segmentation with U-Net

At each step of a decoder, we have the following operations.

- Up-sampling as implemented by transposed convolution followed by activation.
- To retrieve precision at object borders, the output of up-sampling must be concatenated with the result of a corresponding layer at the encoder.
- The result of that concatenation is then processed by a convolutional layer to reduce the number of channels.

Transposed Convolution 2D

The transpose convolution between an image with (N_x, N_y) pixels and a kernel of size (K_x, K_y), padding (P_x, P_y) and strides (S_x, S_y), is an image with (O_x, O_y) pixels, where $O_* = (N_* - 1)S_* + K_* - 2P_*$.

Let's play with it in ▸ (TRANSPOSE CONVOLUTION) and understand its arithmetic in the next slide.

Transposed Convolution 2D

Input 2x2

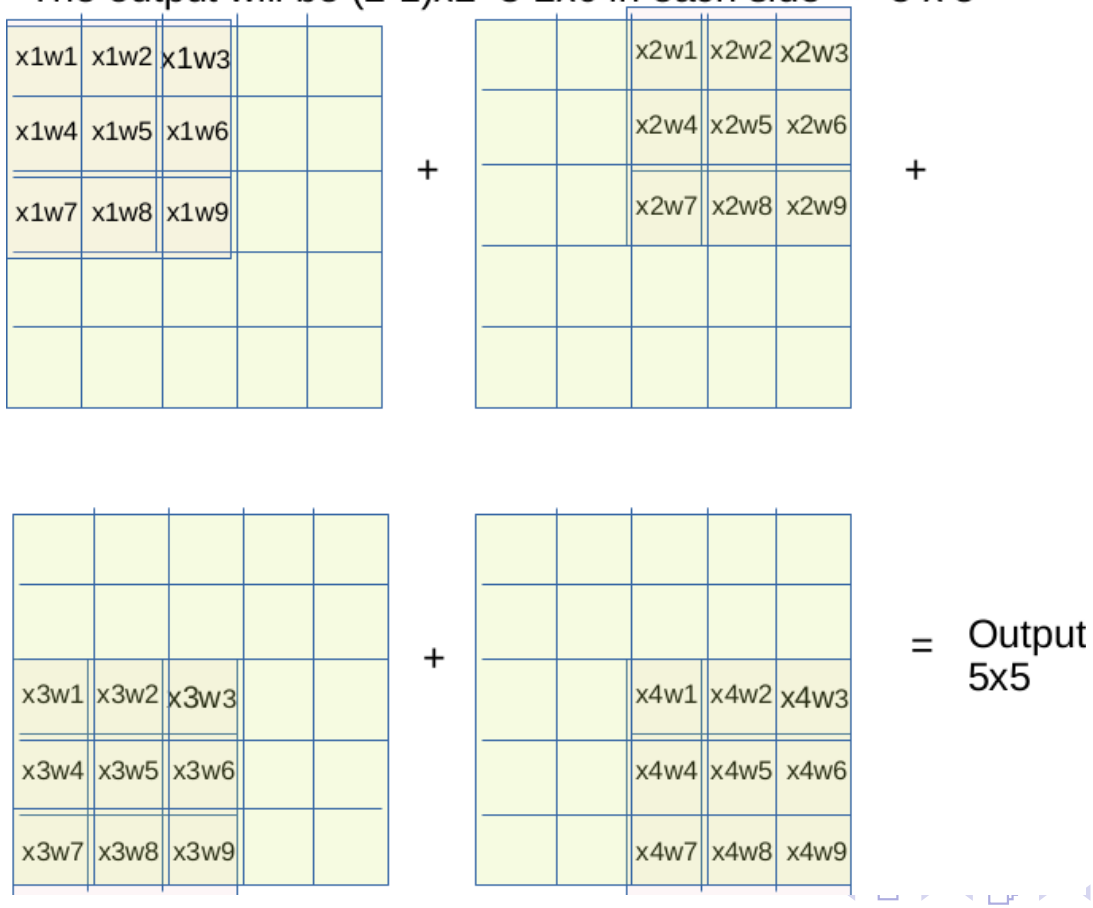
X1	X2
X3	X4

Kernel 3x3

w1	w2	w3
w4	w5	w6
w7	w8	w9

Padding = (0,0)
Strides = (2,2)

The output will be (2-1)x2+3-2x0 in each side => 5 x 5



Playing with U-Net

Now, let's play with U-Net ▸ (SEMANTIC SEGMENTATION)

[1] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár.

Microsoft coco: Common objects in context, 2015.

[2] Raia Hadsell, Sumit Chopra, and Yann Lecun.

Dimensionality reduction by learning an invariant mapping.

In *Compute Vision and Pattern Recognition (CVPR'06)*, pages 1735 – 1742, 02 2006.

[3] Daniel Rho, TaeSoo Kim, Sooill Park, Jaehyun Park, and JaeHan Park.

Understanding contrastive learning through the lens of margins, 2023.

[4] Yingjie Tian, Duo Su, Stanislao Lauria, and Xiaohui Liu.

Recent advances on loss functions in deep learning for computer vision.

Neurocomputing, 497:129–158, 2022.

[5] Ravpreet Kaur and Sarbjeet Singh.

A comprehensive review of object detection with deep learning. *Digital Signal Processing*, 132:103812, 2023.