MO434 - Deep Learning
Applications in Image Analysis - Part I

Alexandre Xavier Falcão

Institute of Computing - UNICAMP

afalcao@ic.unicamp.br

Introduction

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

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

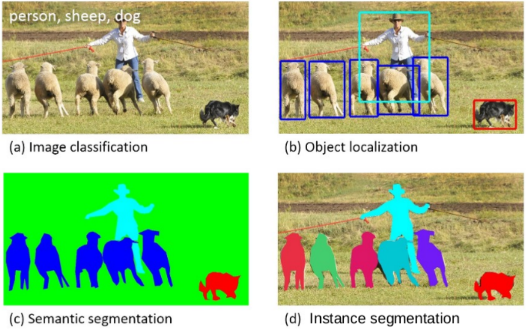
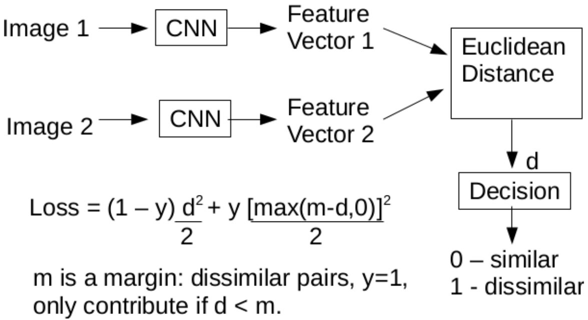


Figure extracted from [1].

Introduction

So far, we have used CNNs to build predictive models. We will now learn how to build contrastive models based on CNNs.



A contrastive model indicates how similar two inputs are [2].

Introduction

After defining a project for this course based on contrastive models, we will understand how to

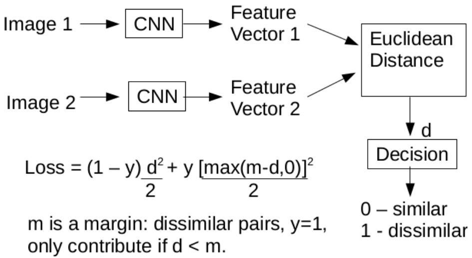
- address object detection using different strategies,
- build fully convolutional models and employ them for
- semantic and instance segmentation.

Agenda

Part I of applications in image analysis will cover

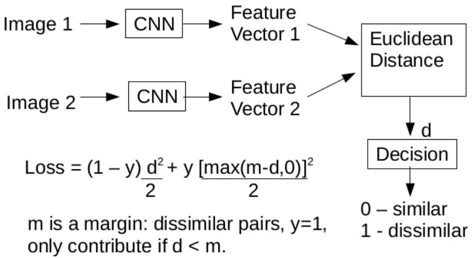
- Building contrastive models.
- The project of this course.
- Strategies for object detection.

Building contrastive models



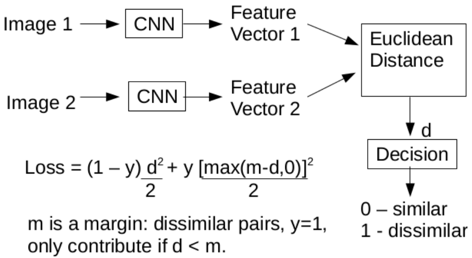
- You may use a pretrained backbone or train a model from scratch, as long as it outputs a feature space suitable for Euclidean distance computation.

Building contrastive models



- You may use a pretrained backbone or train a model from scratch, as long as it outputs a feature space suitable for Euclidean distance computation.
- Recall that dense layers are important to reduce feature spaces.

Building contrastive models



- You may use a pretrained backbone or train a model from scratch, as long as it outputs a feature space suitable for Euclidean distance computation.
- Recall that dense layers are important to reduce feature spaces.
- Let's see one example of few-shot contrastive learning for face verification. (CONTRASTIVE LEARNING).

The project of this course

- Investigate the literature of contrastive learning and try to improve the above notebook for face verification.

The project of this course

- Investigate the literature of contrastive learning and try to improve the above notebook for face verification.
- Evaluate and discuss the role of the hyperparameters contrastive threshold, margin and dimension of the feature space used for distance computation.

The project of this course

- Investigate the literature of contrastive learning and try to improve the above notebook for face verification.
- Evaluate and discuss the role of the hyperparameters contrastive threshold, margin and dimension of the feature space used for distance computation.
- Use some non-linear projection technique to evaluate the impact of your changes in class separation.

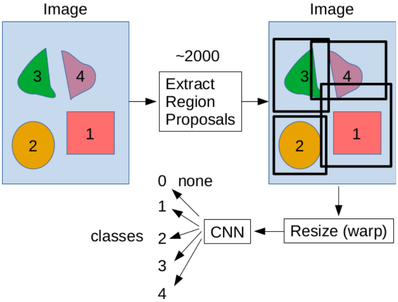
The project of this course

- Investigate the literature of contrastive learning and try to improve the above notebook for face verification.
- Evaluate and discuss the role of the hyperparameters contrastive threshold, margin and dimension of the feature space used for distance computation.
- Use some non-linear projection technique to evaluate the impact of your changes in class separation.
- Can you improve face verification by using a pretrained model (VGG or ResNet) as backbone?

The project of this course

- Investigate the literature of contrastive learning and try to improve the above notebook for face verification.
- Evaluate and discuss the role of the hyperparameters contrastive threshold, margin and dimension of the feature space used for distance computation.
- Use some non-linear projection technique to evaluate the impact of your changes in class separation.
- Can you improve face verification by using a pretrained model (VGG or ResNet) as backbone?
- Finally, repeat this study using the corel dataset, build a predictive model using the resulting features and compare it with your previous solutions.

R-CNN (Region-based CNN) is a popular approach for object detection, which relies on the classification of subimages (candidate regions) extracted from different locations in a given image (`https://paperswithcode.com/method/r-cnn`).



One may slide windows of object-based sizes (anchor boxes) or extract regions from components of a hierarchical segmentation [3]. Selective search [4] follows the second strategy (SELECTIVE SEARCH).

- Region proposal networks adopt the sliding-window strategy and train a CNN to select regions that contain objects.

Strategies for object detection

- Region proposal networks adopt the sliding-window strategy and train a CNN to select regions that contain objects.

- Overlapping regions might be classified as containing a same object, leading to the non-maximum suppression problem, whose solution substitutes them by the most likely one.

Strategies for object detection

- Region proposal networks adopt the sliding-window strategy and train a CNN to select regions that contain objects.

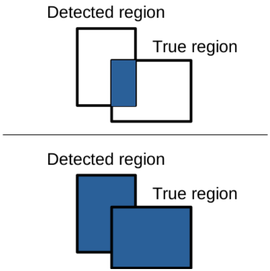
- Overlapping regions might be classified as containing a same object, leading to the non-maximum suppression problem, whose solution substitutes them by the most likely one.

- Training set preparation is challenging, since one has to decide which regions contain each class based on a percentage of object pixels inside the region.

Strategies for object detection

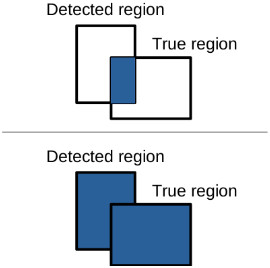
- Region proposal networks adopt the sliding-window strategy and train a CNN to select regions that contain objects.
- Overlapping regions might be classified as containing a same object, leading to the non-maximum suppression problem, whose solution substitutes them by the most likely one.
- Training set preparation is challenging, since one has to decide which regions contain each class based on a percentage of object pixels inside the region.
- Efficiency is low, since the CNN has to process all extracted regions.

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



Strategies for object detection

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

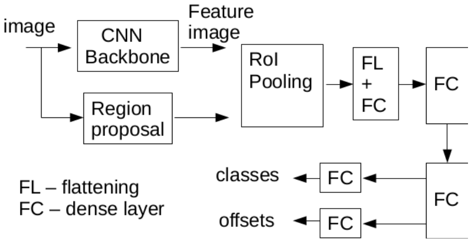


- Precision is the number of true positives (bounding boxes that led to correct prediction) divided by the sum of true positives and false negatives.

- 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).

Strategies for object detection

Fast R-CNN speeds up the process as follows (`https://paperswithcode.com/method/fast-r-cnn`).



Regions are extracted from the backbone's feature map and region warping is substituted by ROI pooling. Two FC layers predict classes and offsets of the regions.

For VGG-16, feature maps are 14×14 pixels, shrinking $14/224$ the input images. ROI pooling identifies the region in that map, crops and resizes it into a 7×7 map.

Strategies for object detection

Input

0.4	0.32	0.01	0.42	0.68	0.67	0.01	0.07
0.54	0.66	0.81	0.18	0.01	0.42	0.46	0.19
0	0.38	0.31	1	0.06	0.1	0.48	0.91
0.63	0.91	0.31	0.41	0.4	0.85	0.56	0.39
0.09	0.42	0.5	0.22	0.8	0.98	0.13	0.88
0.99	0.51	0.16	0.68	0.9	0.23	0.57	0.93
0.8	0.4	0.06	0.48	0.52	0.88	0.2	0.17
0.64	0.92	0.07	0.73	0.24	0.2	0.79	0.16

Region proposal

0.4	0.32	0.01	0.42	0.68	0.67	0.01	0.07
0.54	0.66	0.81	0.18	0.01	0.42	0.46	0.19
0	0.38	0.31	1	0.06	0.1	0.48	0.91
0.63	0.91	0.31	0.41	0.4	0.85	0.56	0.39
0.09	0.42	0.5	0.22	0.8	0.98	0.13	0.88
0.99	0.51	0.16	0.68	0.9	0.23	0.57	0.93
0.8	0.4	0.06	0.48	0.52	0.88	0.2	0.17
0.64	0.92	0.07	0.73	0.24	0.2	0.79	0.16

Dividing the region into sections

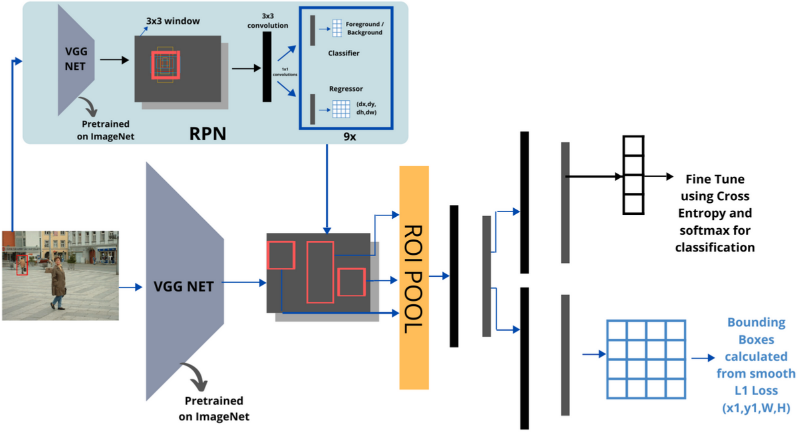
0.4	0.32	0.01	0.42	0.68	0.67	0.01	0.07
0.54	0.66	0.81	0.18	0.01	0.42	0.46	0.19
0	0.38	0.31	1	0.06	0.1	0.48	0.91
0.63	0.91	0.31	0.41	0.4	0.85	0.56	0.39
0.09	0.42	0.5	0.22	0.8	0.98	0.13	0.88
0.99	0.51	0.16	0.68	0.9	0.23	0.57	0.93
0.8	0.4	0.06	0.48	0.52	0.88	0.2	0.17
0.64	0.92	0.07	0.73	0.24	0.2	0.79	0.16

Max pooling

0.91	0.98
0.99	0.88

If you want to output a 2×2 region from a proposed region with 5×7 pixels, ROI pooling divides the region into 2×2 sections for max-pooling inside each section.

Strategies for object detection

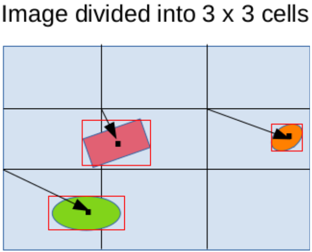


Faster R-CNN (`https://paperswithcode.com/paper/faster-r-cnn-towards-real-time-object`) substitutes selective search, used in R-CNN and Fast R-CNN, by a Region Proposal Network (RPN).

Strategies for object detection

You Only Look Once (YOLO) further speeds up detection. Assuming one object per cell, we can

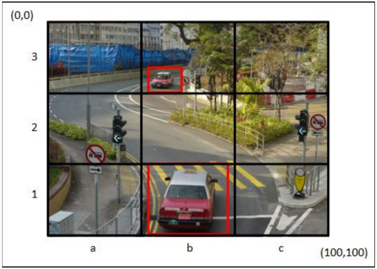
- divide each image into $N \times N$ cells,
- identify which cells contain the center of the ground-truth bounding box, and
- train a CNN to output $N \times N$ estimates of class, proportional size and relative offset of the objects in an image.



Strategies for object detection

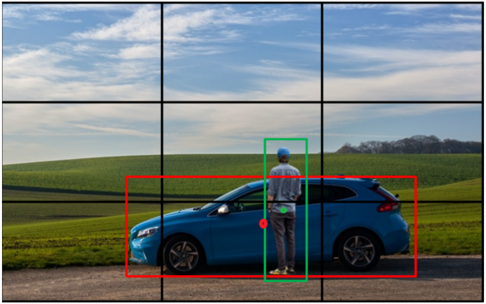
You Only Look Once (YOLO) further speeds up detection. Assuming one object per cell, we can

- divide each image into $N \times N$ cells,
- identify which cells contain the center of the 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.



Strategies for object detection

Again, papers and codes of YOLO-based versions can be obtained from `https://paperswithcode.com/search?q_meta=&q_type=&q=YOLO`.

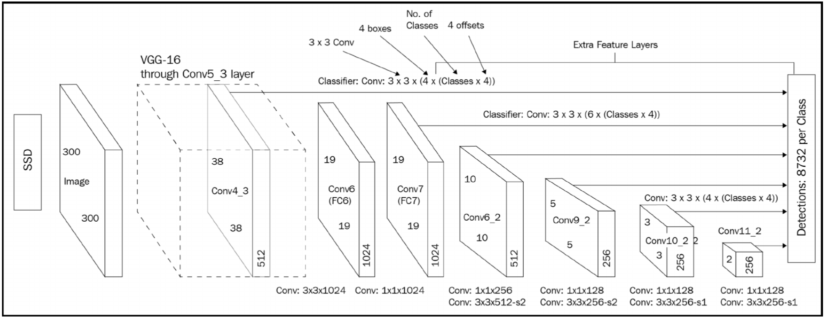


For multiple objects per cell, one can define bounding boxes of different aspect ratios to represent distinct objects (anchor boxes) inside each cell.

Strategies for object detection

Single Shot Multibox Detector (SSD) differs from YOLO by

- discretizing the number of possible bounding boxes (scales and aspect ratios) and
- using the last backbone layers with additional ones to cope with multi-scale multi-object detection per cell.



[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] F.L. Galvão, S.J.F. Guimarães, and A.X. Falcão.
Image segmentation using dense and sparse hierarchies of superpixels.
Pattern Recognition, 108:107532, 2020.

[4] J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers, and et al.
Selective search for object recognition.
Int J Comput Vis, 104:154–171, 2013.