How Object Detection has been transformed by Deep Learning

Paul Blondel, PhD

MIS-Heudyasic, France

November 9th, 2019

1. Introduction
2. How to detect objects in an image?
3. Training the blackbox with Machine Learning
4. Deep learning
5. Conclusion

1 Introduction

2 How to detect objects in an image?

3 Training the blackbox with Machine Learning

4 Deep learning

5 Conclusion

Object Detection?

- GOAL: **detect objects** in images or video frames
- Can sometimes **detect different types** of object **simultaneously**

Object Detection:

A solution to treat the **ever growing** amount of images and video frames

Object detection is **not so easy**:

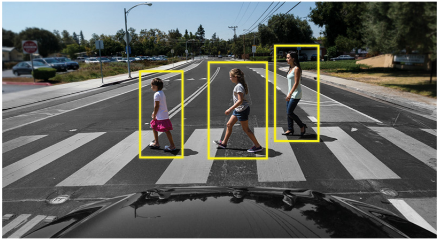
- objects can have **multiple scales**
- objects can be **partially hidden**
- object types can be **look similar**
  - ex: lions and cats
- a same object type can have **different textures, colors, etc**
  - ex: human people wearing **different clothes**
- objects can have **multiple orientations and postures**
- etc

Object detection can have other constraints:

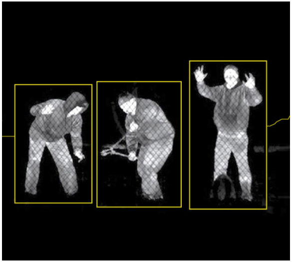
- **real-time** detection
- work on **embedded systems**
  - ex: cars, UAVs, etc.
- **weather conditions**
- **night conditions**
- etc.


Applications:

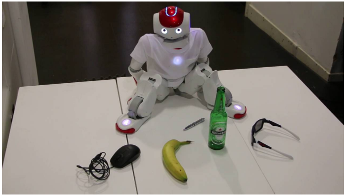
- Advanced Driver Assistance System (ADAS)



- Video surveillance:



- Robots



- Face detection

- ...

1 Introduction

2 How to detect objects in an image?

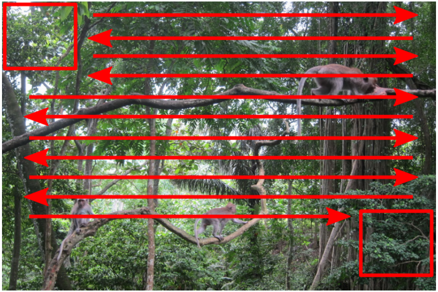
3 Training the blackbox with Machine Learning

4 Deep learning

5 Conclusion

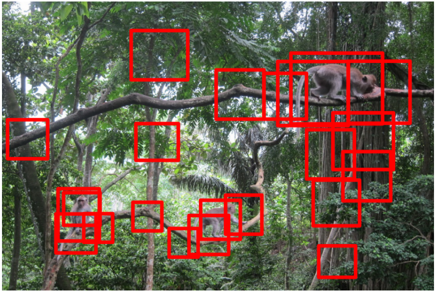
How to find these monkeys?

Searching at multiple locations



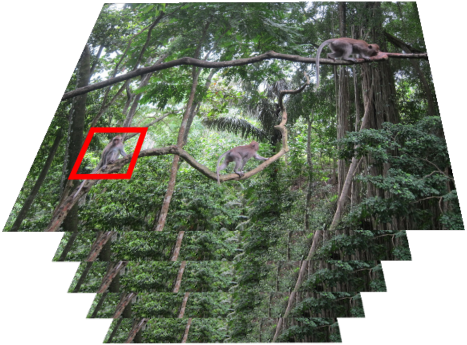
- **Sliding Window:**
  Exhaustive scan of the image

Searching at multiple locations



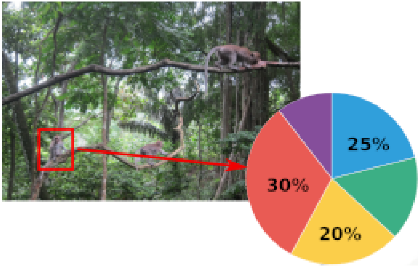
- Sliding Window:
  Exhaustive scan of the image
- **Generate region proposals:**
  Info-rich regions are proposed

Searching at multiple scales



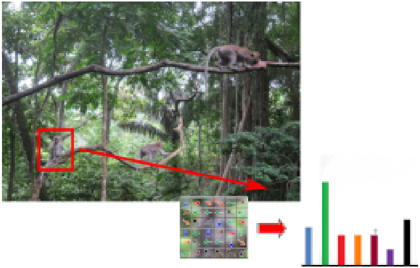
- Analysis window
  - Fixed size
- Image pyramid
  - Down-scaled levels for big objects
  - Up-scaled levels for small objects

Finding clues



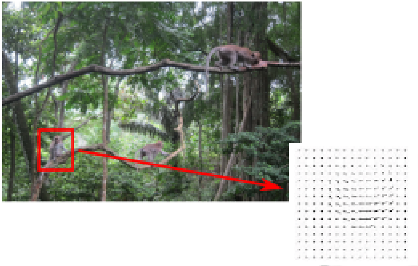
- Visual features
  - **Colors**

Finding clues



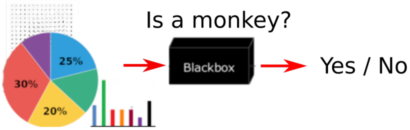
- Visual features
  - Colors
  - **Shapes**

Finding clues



- Visual features
  - Colors
  - Shapes
  - **Movements**
  - **Etc.**

Analyze the collected clues and decide!



- Classify
  - Visual features of a monkey...
  - ...or not
- Deep learning
  - All these steps may be combined

1 Introduction

2 How to detect objects in an image?

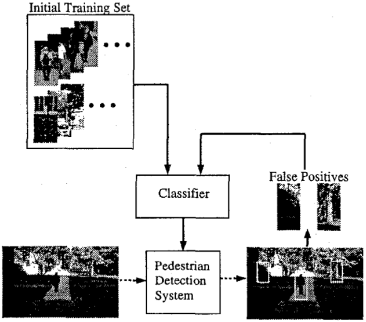
3 Training the blackbox with Machine Learning

4 Deep learning

5 Conclusion

How to train the blackbox (classifier)? With Machine Learning!

- Papageorgiou et al: Training the classifier with a SVM algorithm

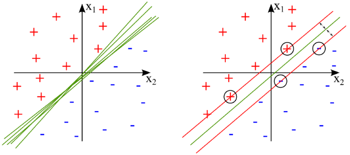


At first, to **train the blackbox** we **need a lot of image examples**:

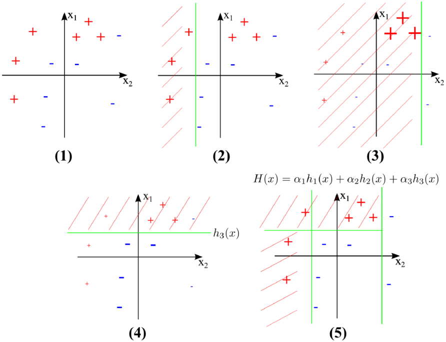
- images of **object** (ex: images of people)
- images of **random background**

Then, we can **train the blackbox** with a Machine Learning algorithm:

- Support Vector Machine (SVM)

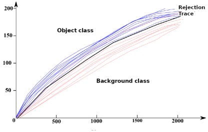


- Boosting



Before 2014, there have been some **improvements**:

- New classifiers (Soft-Cascade Boosting, Latent-SVM, etc))



- New ways of collecting clues/features (HOG, ICF, ACF, etc)

With this approach ...

NO real dramatic improvements... after 2014: Deep learning!

1 Introduction

2 How to detect objects in an image?

3 Training the blackbox with Machine Learning

4 Deep learning

5 Conclusion

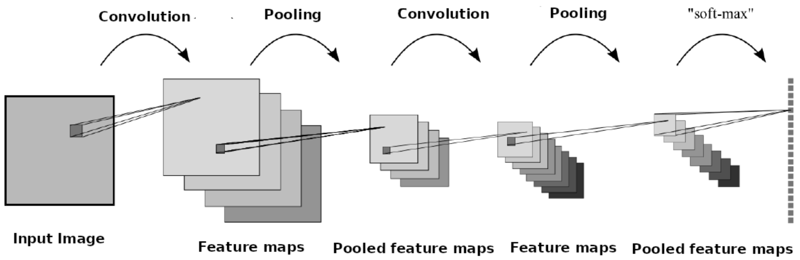
The **COME BACK of Artificial Neural Networks**:

- Artificial Neural Networks (ANN) **exist for a very long time** (50's)
- But: the more recent SVM beat ANNs for a while
- Among all ANN: **CNN** is the **most suitable for Computer Vision**
- Deep learning: deep means a network with more than 4/5 layers

The emergence of Deep learning is due to:

- New network **learning approaches**
- **Fixing some problems** (vanishing gradient, etc.)
- **More data** available everywhere
- More and more **powerful computers**

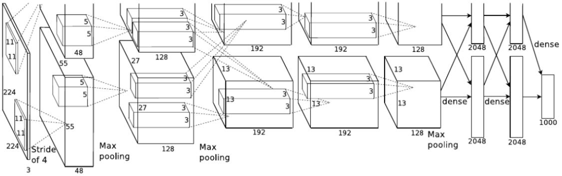
The Convolutional Neural Network (CNN) is as follow:



- Convolution: **local pixels** are **connected to the same pool node**
- Pooling: **features** computed in the convolution layers **are aggregated** (max or average over a pool)
- Images = many pixels, thanks to CNN: we **don't need a tremendeous number of connections**

In 2012 the first Deep CNN:

- It won the "ImageNet" contest (1.6 millions images):
  - Can reconize 1000 object types
  - 37.5% error rate (previous best: 45.1%)



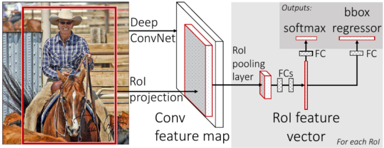
BUT:

It's NOT OBJECT DETECTION it's OBJECT CLASSIFICATION

A series of improvements:

- In 2014, R-CNN: it can detect objects!
  - Generate **region proposals** to search objects (Selective Search)
  - Use Deep CNN to collect clues/features
  - 53.7% of mAP PASCAL 2010 (previous best: 33.4%)

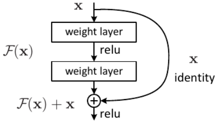
- In 2015, Fast R-CNN: faster
  - **All clues/features are computed once!**



- The same year, Faster R-CNN: even faster
  - The network itself **generate region proposals**!

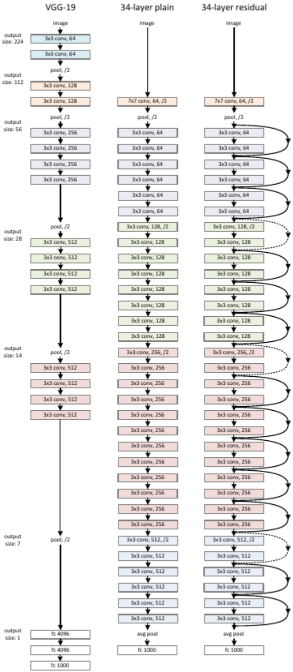
So far, "deep" meant 4/5 layers, but **residual learning** permits more:

- This is a "residual mapping" $(F(x) + X)$



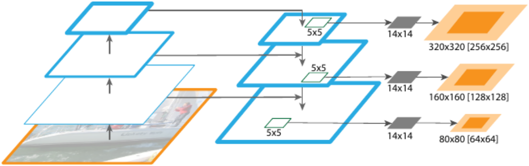
- The architecture of the network can have >100 layers with this!
- 19.38% error rate on ImageNet! (previous best: 37.5%)

Residual learning means better detection performance!


VGG-19
34-layer plain
34-layer residual

In 2017, FPN: more robust to object sizes!

- There is an **image pyramid INSIDE the network**



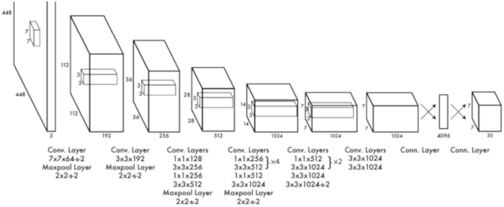
- Faster R-CNN VS FPN: Improve by 2% in AP (COCO dataset)

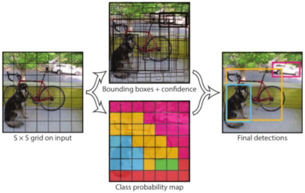
Detection performance

Accuracy gets improved again!

In 2016, YOLO: an alternative approach

- Here: it is a **regression problem**
  - all combined: **search**, **extract clues/features** AND **infer**
  - so, there is **no need to generate region proposals**
- Learn the context as well (more general)
- More **easy to train**
- Much faster: 45 FPS on Titan GPU!

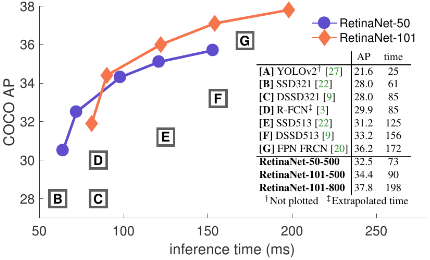




In 2017, RetinaNet: Regression outperformed classification!

- A new training loss function to optimize:
  - Called Focal Loss
  - **Focus** the learning on **hard background images**
- Outperforms all classification approaches on COCO (speed VS AP)!

$$\mathrm{FL}(p_t) = -(1-p_t)^{\gamma}\log(p_t).$$



A great step towards a all-in-one network object detector

1 Introduction

2 How to detect objects in an image?

3 Training the blackbox with Machine Learning

4 Deep learning

5 Conclusion

To conclude:

- Former Machine Learning approaches for OD: obsolete
- Performances improved thanks to Deep Learning
- Year after year, Deep Learning-based Object Detection becomes ...
  - ... simpler (one step training, etc.)
  - ... more accessible (cheaper and cheaper powerful GPU, etc.)
  - ... more accurate (new optimizations, etc.)
  - ... speeder.
- A clear trend: one unique network for all detection steps

In 2018, YOLOv3: two to three times faster than RetinaNet, with same accuracy ...

The course continue...