**State-of-art about machine learning approaches and study of a specific deep learning architecture**

1. **The Rise of deep learning :**

In this section we will discuss a general overview of the history of the field and it’s recent progress and result including the application on computer vision tasks

1. **History of deep learning**
   * In this section we will discuss the. What is deeplearning and what are different branch of the field as well as history of the field including historical figures(fathers of the field ) , benefits that came from other fields such as neuroscience
2. **The dramatic progress in the field including state of the-art models and solutions**
   * In this section we will start from the first models of deeplearning until state of the art in details, we will end it by modern applications of the DL techniques emphasizing on Computer vision and image processing
3. **Future of the field and it’s relationship with the rise of computation and other fields (Cognitive science , quantum Computers ..)**
   * potential applications and potential exploits of other fields in order to improve current models
   * latest breakthrough
4. **Computer vision and deep learning**
   1. General history of computer vision and progress until reaching deeepl learning basesd technieus . Use Bag of words vs deep learning paper
5. **Comparison between classical methods vs deep learning bases approaches**
   1. Table to compares every techniques
6. **Conclusio**

**The Rise of deep learning**

Inventors have long dreamed of creating machines that think. This desire dates back to at least the time of ancient Greece. The mythical figures Pygmalion, Daedalus, and Hephaestus may all be interpreted as legendary inventors, and Galatea, Talos, and Pandora may all be regarded as artificial life (Ovid and Martin, 2004; Sparkes, 1996; Tandy, 1997).

When programmable computers were first conceived, people wondered whether such machines might become intelligent, over a hundred years before one was built (Lovelace, 1842). Today, artificial intelligence (AI) is a thriving field with many practical applications and active research topics. We look to intelligent software to automate routine labor, understand speech or images, make diagnoses in medicine and support basic scientific research.

In the early days of artificial intelligence, the field rapidly tackled and solved problems that are intellectually difficult for human beings but relatively straight- forward for computers—problems that can be described by a list of formal, math- ematical rules. The true challenge to artificial intelligence proved to be solving the tasks that are easy for people to perform but hard for people to describe formally—problems that we solve intuitively, that feel automatic, like recognizing spoken words or faces in images.

This solution is to allow computers to learn from experience and understand the world in terms of a hierarchy of concepts, with each concept defined in terms of its relation to simpler concepts. By gathering knowledge from experience, this approach avoids the need for human operators to formally specify all of the knowledge that the computer needs. The hierarchy of concepts allows the computer to learn complicated concepts by building them out of simpler ones. If we draw a graph showing how these concepts are built on top of each other, the graph is deep, with many layers. For this reason, we call this approach to AI deep learning.

A close up of a mans face

Description automatically generated

Figure 1.4: A Venn diagram showing how deep learning is a kind of representation learning, which is in turn a kind of machine learning, which is used for many but not all approaches to AI. Each section of the Venn diagram includes an example of an AI technology.

The term *Deep Learning* was introduced to the machine learning community by [Rina Dechter](https://en.wikipedia.org/wiki/Rina_Dechter" \o "Rina Dechter) in 1986 and to [artificial neural networks](https://en.wikipedia.org/wiki/Artificial_Neural_Networks" \o "Artificial Neural Networks) by Igor Aizenberg and colleagues in 2000, in the context of [Boolean](https://en.wikipedia.org/wiki/Boolean_network" \o "Boolean network) threshold neurons.

The first general, working learning algorithm for supervised, deep, feedforward, multilayer [perceptrons](https://en.wikipedia.org/wiki/Perceptron" \o "Perceptron) was published by [Alexey Ivakhnenko](https://en.wikipedia.org/wiki/Alexey_Ivakhnenko" \o "Alexey Ivakhnenko) and Lapa in 1967 A 1971 paper described already a deep network with 8 layers trained by the [group method of data handling](https://en.wikipedia.org/wiki/Group_method_of_data_handling" \o "Group method of data handling) algorithm.

Other deep learning working architectures, specifically those built for [computer vision](https://en.wikipedia.org/wiki/Computer_vision" \o "Computer vision), began with the [Neocognitron](https://en.wikipedia.org/wiki/Neocognitron" \o "Neocognitron) introduced by [Kunihiko Fukushima](https://en.wikipedia.org/wiki/Kunihiko_Fukushima" \o "Kunihiko Fukushima) in 1980.[[34]](https://en.wikipedia.org/wiki/Deep_learning" \l "cite_note-FUKU1980-34) In 1989, [Yann LeCun](https://en.wikipedia.org/wiki/Yann_LeCun" \o "Yann LeCun) et al. applied the standard backpropagation algorithm, which had been around as the reverse mode of [automatic differentiation](https://en.wikipedia.org/wiki/Automatic_differentiation) since 1970 to a deep neural network with the purpose of [recognizing handwritten ZIP codes](https://en.wikipedia.org/wiki/Handwriting_recognition) on mail. While the algorithm worked, training required 3 days.

In 2012, a team led by George E. Dahl won the "Merck Molecular Activity Challenge" using multi-task deep neural networks to predict the [biomolecular target](https://en.wikipedia.org/wiki/Biomolecular_target" \o "Biomolecular target) of one drug In 2014, Hochreiter's group used deep learning to detect off-target and toxic effects of environmental chemicals in nutrients, household products and drugs and won the "Tox21 Data Challenge" of [NIH](https://en.wikipedia.org/wiki/NIH" \o "NIH), [FDA](https://en.wikipedia.org/wiki/FDA" \o "FDA) and [NCATS](https://en.wikipedia.org/wiki/National_Center_for_Advancing_Translational_Sciences).

Significant additional impacts in image or object recognition were felt from 2011 to 2012. Although CNNs trained by backpropagation had been around for decades, and GPU implementations of NNs for years, including CNNs, fast implementations of CNNs with max-pooling on GPUs in the style of Ciresan and colleagues were needed to progress on computer vision In 2011, this approach achieved for the first time superhuman performance in a visual pattern recognition contest. Also in 2011, it won the ICDAR Chinese handwriting contest, and in May 2012, it won the ISBI image segmentation contest  Until 2011, CNNs did not play a major role at computer vision conferences, but in June 2012, a paper by Ciresan et al. at the leading conference CVPR[[](https://en.wikipedia.org/wiki/Deep_learning" \l "cite_note-:9-4) showed how max-pooling CNNs on GPU can dramatically improve many vision benchmark records. In October 2012, a similar system by Krizhevsky et al won the large-scale [ImageNet competition](https://en.wikipedia.org/wiki/ImageNet_competition" \o "ImageNet competition) by a significant margin over shallow machine learning methods. In November 2012, Ciresan et al.'s system also won the ICPR contest on analysis of large medical images for cancer detection, and in the following year also the MICCAI Grand Challenge on the same topic.[  In 2013 and 2014, the error rate on the ImageNet task using deep learning was further reduced, following a similar trend in large-scale speech recognition. The [Wolfram](https://en.wikipedia.org/wiki/Stephen_Wolfram) Image Identification project publicized these improvements.[

Image classification was then extended to the more challenging task of [generating descriptions](https://en.wikipedia.org/wiki/Automatic_image_annotation) (captions) for images, often as a combination of CNNs and LSTMs.

Some researchers assess that the October 2012 ImageNet victory anchored the start of a "deep learning revolution" that has transformed the AI industry

In March 2019, [Yoshua Bengio](https://en.wikipedia.org/wiki/Yoshua_Bengio" \o "Yoshua Bengio), [Geoffrey Hinton](https://en.wikipedia.org/wiki/Geoffrey_Hinton" \o "Geoffrey Hinton) and [Yann LeCun](https://en.wikipedia.org/wiki/Yann_LeCun" \o "Yann LeCun) were awarded the [Turing Award](https://en.wikipedia.org/wiki/Turing_Award" \o "Turing Award) for conceptual and engineering breakthroughs that have made deep neural networks a critical component of computing

**A screenshot of a cell phone

Description automatically generated**

Figure 1.5: Flowcharts showing how the different parts of an AI system relate to each other within different AI disciplines. Shaded boxes indicate components that are able to learn from data.

**Computer vision and deep learning**

The past decade has seen the rise of different approaches for pattern-recognition task in computer vision. A study has enabled us to identify two methods: the Bag of Features (BoF) and the deep learning technique with the use of convolutional neural network (CNN).

The BoF methods have been applied to image classification, object detection, image retrieval, and even visual localization for robots. BoF approaches are characterized by the use of an orderless collection of image features. Lacking any structure or spatial information, it is perhaps surprising that this choice of image representation would be powerful enough to match or exceed state-of-the-art performance in many of the applications to which it has been applied. Due to its simplicity and performance, the Bag of Features approach has become well- established in the field.

CNN used in deep learning learn hierarchical layers of representation from input in order to perform image classification [1-2]. Recently, these deep architectures have demonstrated impressive, state-of-the-art, and sometimes human-competitive results on many pattern recognition tasks, especially vision classification problems [3-5].

Diffrent problems in Computer vision :

1. **Image Classification :**

Image Classification problem is the task of **assigning an input image one label from a fixed set of categories**. This is one of the core problems in CV that, despite its simplicity, has a large variety of practical applications , ImageNet is one of the famous competition Benchmark that evaluates solutions proposed by researches around the world and their definition of Image classification :

“ For each image, algorithms will produce a list of at most 5 object categories in the descending order of confidence. The quality of a labeling will be evaluated based on the label that best matches the ground truth label for the image. The idea is to allow an algorithm to identify multiple objects in an image and not be penalized if one of the objects identified was in fact present, but not included in the ground truth. For each image, an algorithm will produce 5 labels lj,j=1,…,5lj,j=1,…,5. The ground truth labels for the image are gk,k=1,…,ngk,k=1,…,n with n classes of objects labeled. The error of the algorithm for that image would be

e=1n⋅∑kminjd(lj,gk),e=1n⋅∑kminjd(lj,gk),

where d(x,y)=0d(x,y)=0 if x=yx=y and 1 otherwise. The overall error score for an algorithm is the average error over all test images. Note that for this version of the competition, n=1n=1, that is, one ground truth label per image. “

A screenshot of a cell phone

Description automatically generated

**2. Object Localization (Loc)**

In fact, this is the most confusing task when I first look at ImageNet challenges.

This is a sort of intermediate task in between other two ILSRVC tasks, image classification and object detection. In image classification you have to **assign a single label to an image corresponding to the “main” object** (eventually, the image can contain multiple objects). The classification + localization requires also to **localize a single instance of this object**, even if the image contains multiple instances of it. This task is also called “single-instance localization”

The definition of localization in ImageNet is:

In this task, an algorithm will produce 5 class labels lj,j=1,…,5lj,j=1,…,5 and 5 bounding boxes bj,j=1,…5bj,j=1,…5, one for each class label. The ground truth labels for the image are gk,k=1,…,ngk,k=1,…,n with n classes labels. For each ground truth class label gkgk, the ground truth bounding boxes are zkm,m=1,…Mk,zkm,m=1,…Mk, where MkMk is the number of instances of the kthkth object in the current image. The error of the algorithm for that image would be

e=1n⋅∑kminjminMkmmax{d(lj,gk),f(bj,zkm)},e=1n⋅∑kminjminmMkmax{d(lj,gk),f(bj,zkm)},

where f(bj,zk)=0f(bj,zk)=0 if bjbj and zmkzmk has over 50% overlap, and f(bj,zmk)=1f(bj,zmk)=1 otherwise. In other words, the error will be the same as defined in classification task if the localization is correct(i.e. the predicted bounding box overlaps over 50% with the ground truth bounding box, or **in the case of multiple instances of the same class, with any of the ground truth bounding boxes**), otherwise the error is 1(maximum).

A screenshot of a cell phone

Description automatically generated

### **2.1 Typical solutions & models :**

\* Treat LOC as regression problem

**A screenshot of text

Description automatically generated**

1. Train a classification model (AlexNet, VGG, GoogLeNet);
2. Attach new fully-connected “regression head” to the network;

**A close up of a device

Description automatically generated**

3.Train the regression head only with SGD and L2 loss;

4.At test time use both heads.

A screenshot of a cell phone

Description automatically generated

**3 Object Detection**

Object detection is the process of **finding instances of real-world objects such as faces, bicycles, and buildings in images or videos**. Object detection algorithms typically use extracted features and learning algorithms to recognize instances of an object category. It is commonly used in applications such as image retrieval, security, surveillance, and automated vehicle parking systems

Image Net Formula for object detection :

“ For each image, algorithms will produce a set of annotations (ci,si,bi)(ci,si,bi) of class labels cici, confidence scores sisi and bounding boxes bibi. This set is expected to contain each instance of each of the 200 object categories. Objects which were not annotated will be penalized, as will be duplicate detections (two annotations for the same object instance). The winner of the detection challenge will be the team which achieves first place accuracy on the most object categories.

“

### **3.1 Typical solutions & models :**

**Using sliding Window:**

1. Run classification + regression network at multiple locations on a high-resolution image;
2. Convert fully-connected layers into convolutional layers for efficient computation;
3. Combine classifier and regressor predictions across all scales for final prediction.

Efficient sliding window by converting fully-connected layers into convolutions

Recent advances in object detection are driven by the success of region proposal methods and region-based convolutional neural networks (R-CNNs). Although region-based CNNs were computationally expensive as originally developed in , their cost has been drastically reduced thanks to sharing convolutions across proposals . The latest incarnation, Fast R-CNN, achieves near real-time rates using very deep networks , when ignoring the time spent on region proposals. Now, proposals are the computational bottleneck in state-of-the-art detection systems. Region proposal methods typically rely on inexpensive features and economical inference schemes. Selective Search (SS) , one of the most popular methods, greedily merges superpixels based on engineered low-level features. Yet when compared to efficient detection networks , Selective Search is an order of magnitude slower, at 2s per image in a CPU implementation. EdgeBoxes currently provides the best tradeoff between proposal quality and speed, at 0.2s per image. Nevertheless, the region proposal step still consumes as much running time as the detection network.

One may note that fast region-based CNNs take advantage of GPUs, while the region proposal methods used in research are implemented on the CPU, making such runtime comparisons inequitable. An obvious way to accelerate proposal computation is to re-implement it for the GPU. This may be an effective engineering solution, but re-implementation ignores the down-stream detection network and therefore misses important opportunities for sharing computation. , latest solution for this problem proposed by Microsoft were they introduced novel Region Proposal Networks (RPNs) that share convolutional layers with state-of-the-art object detection networks. By sharing convolutions at test-time, the marginal cost for computing proposals is small (e.g., 10ms per image),

A Region Proposal Network (RPN) takes an image (of any size) as input and outputs a set of rectangular object proposals, each with an objectness score.

Compared to other region proposal classification networks (fast RCNN) which perform detection on various region proposals and thus end up performing prediction multiple times for various regions in a image, YOLO architecture is more like FCNN (fully convolutional neural network) and passes the image (nxn) once through the FCNN and output is (mxm) prediction. This the architecture is splitting the input image in mxm grid and for each grid generation 2 bounding boxes and class probabilities for those bounding boxes. Note that bounding box is more likely to be larger than the grid itself

## 3 .Segmentation

## 3.1 Semseantic segmentation.

Semantic segmentation is the task of classifying each and very pixel in an image into a class as shown in the image below. Here you can see that all persons are red, the road is purple, the vehicles are blue, street signs are yellow etc.

Semantic segmentation is different from instance segmentation which is that different objects of the same class will have different labels as in person1, person2 and hence different colours. The picture below very crisply illustrates the difference between instance and semantic segmentation

**3.4 instance segmentation**

Instance segmentation is a challenging computer vision task that requires the prediction of object instances and their per-pixel segmentation mask. This makes it a hybrid of semantic segmentation and object detection.

**A picture containing clock

Description automatically generated**

Ever since [Mask R-CNN](https://arxiv.org/abs/1703.06870) was invented, the state-of-the-art method for instance segmentation has largely been Mask RCNN and its variants ([PANet](https://arxiv.org/abs/1803.01534), [Mask Score RCNN](https://arxiv.org/abs/1903.00241), etc). It adopts the detect-then-segment approach, first perform object detection to extract bounding boxes around each object instances, and then perform binary segmentation inside each bounding box to separate the foreground (object) and the background.

However, Mask RCNN is quite slow and precludes the use of many real-time applications. In addition, masks predicted by Mask RCNN have fixed resolution and thus are not refined enough for large objects with complex shapes. There has been a wave of studies on single-stage instance segmentation, fueled by the advances in anchor-free object detection methods (such as [CenterNet](https://arxiv.org/abs/1904.07850) and [FCOS](https://arxiv.org/abs/1904.01355). See my [slides](https://docs.google.com/presentation/d/1_dUfxv63108bZXUnVYPIOAdEIkRZw5BR9-rOp-Ni0X0/edit?usp=sharing) for a quick intro into anchor-free object detection). Many of these methods are faster and more accurate than Mask RCNN, as shown in the image below.

A close up of a map

Description automatically generated