Convolutional Neural network

Ngo Minh Nhut

Summer School – FIT@HCMUS

2022

Outline

- Convolutional neural network
- Practice with Google Colab

Problems in computer vision

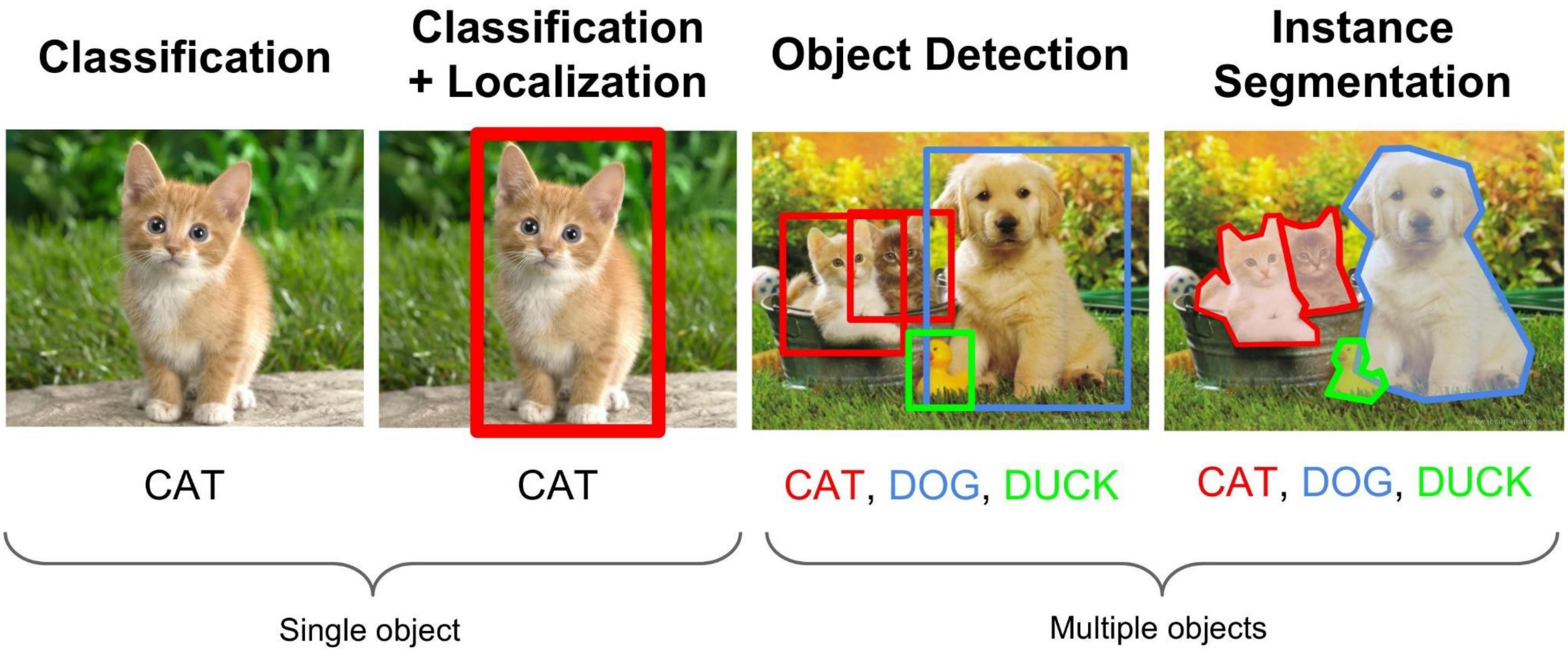
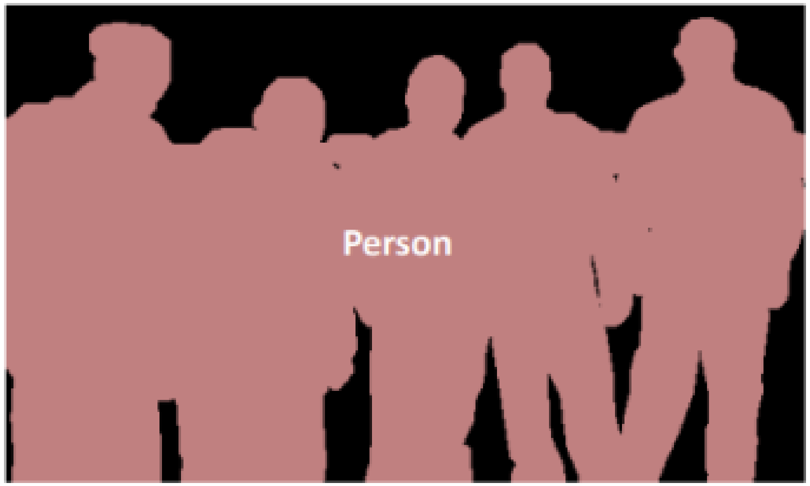


Image segmentation



Semantic Segmentation

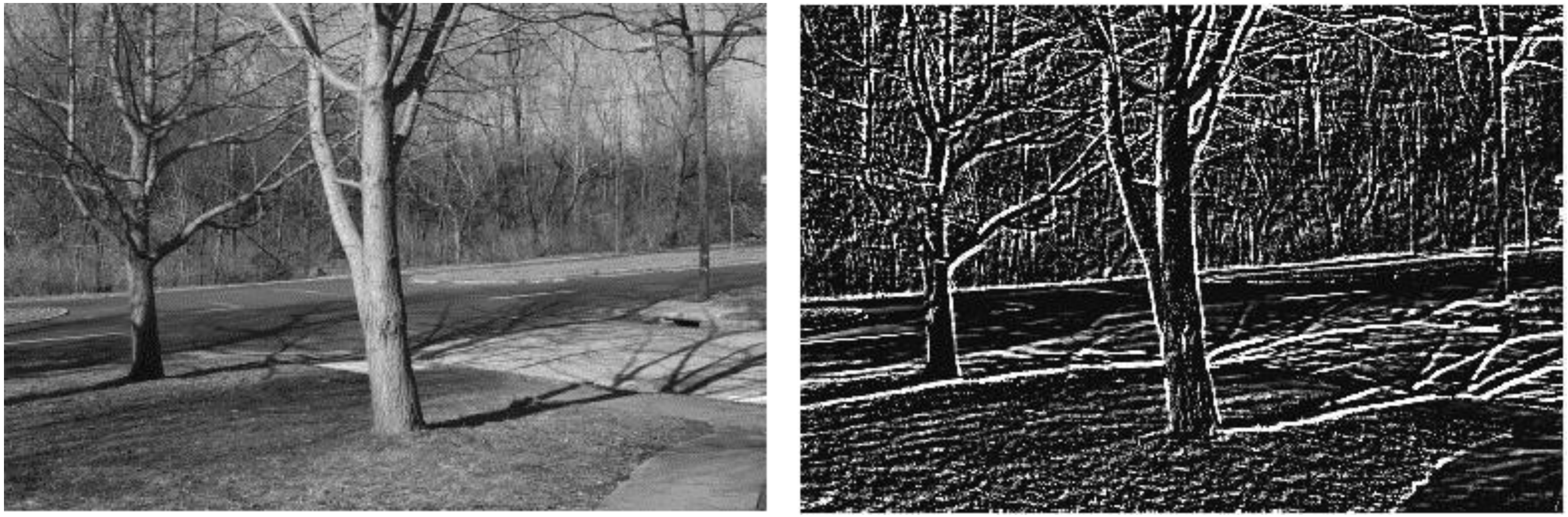
Instance Segmentation

Segmentation techniques

- The pixel values will be different for the objects and the image's background → **Threshold Segmentation**
- **Edge Detection Segmentation -** There is always an edge between two adjacent regions with different grayscale values.
- **Image Segmentation based on Clustering**

Convolutional neural network

- High resolution images contains O(millions) of pixels
- A neural network which can handle that kind of images would also have O(millions) of weight



Convolutional filter

Convolutional neural network

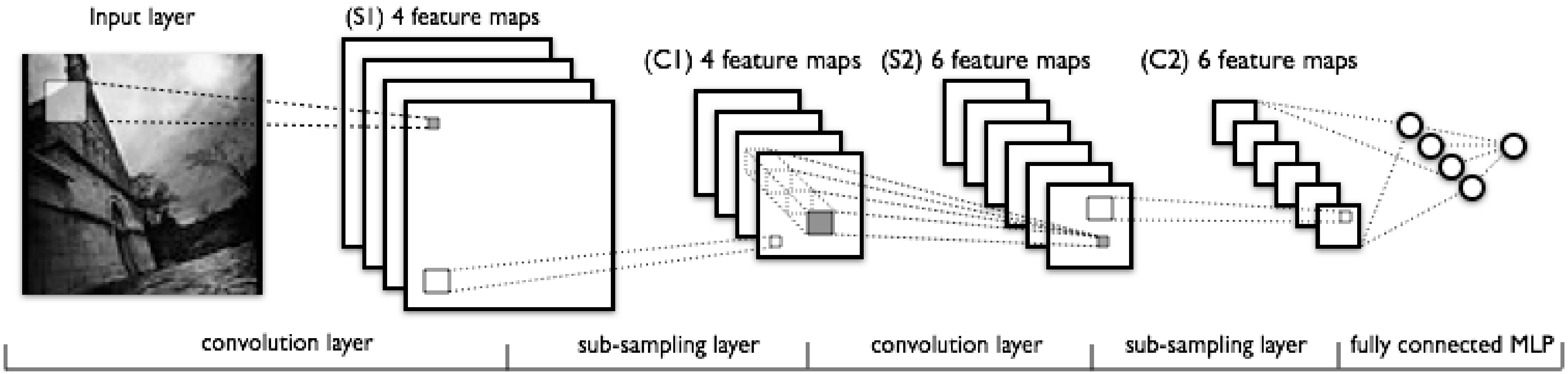
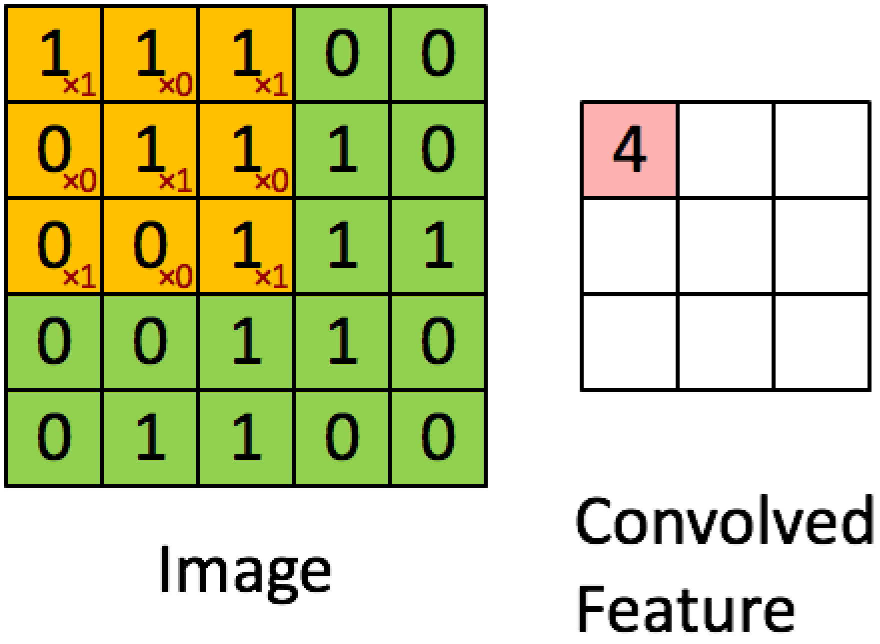


Illustration: Theano documentation

This kind of architecture will learn filters and build an internal representation of the input data using many stacked layers and finally use this representation on a classification task.

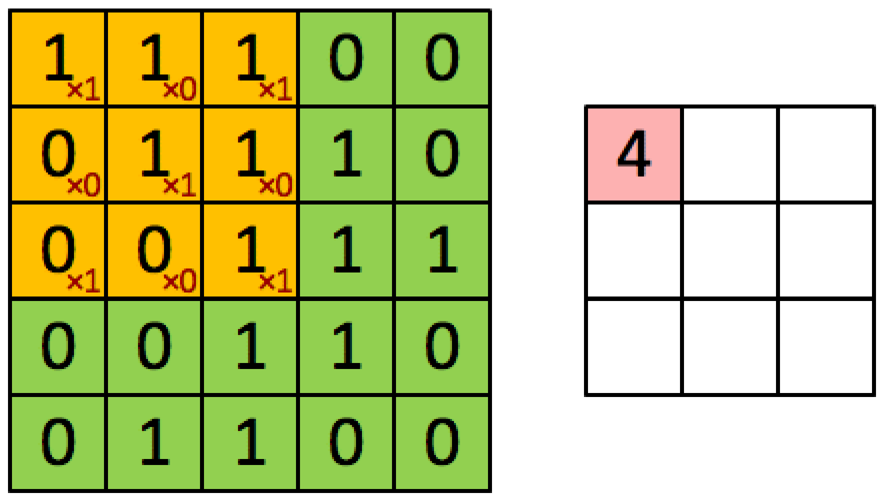
Convolutional neural network

- Filters



Convolutional neural network

- Filters



Image

Convolved Feature

Convolutional neural network

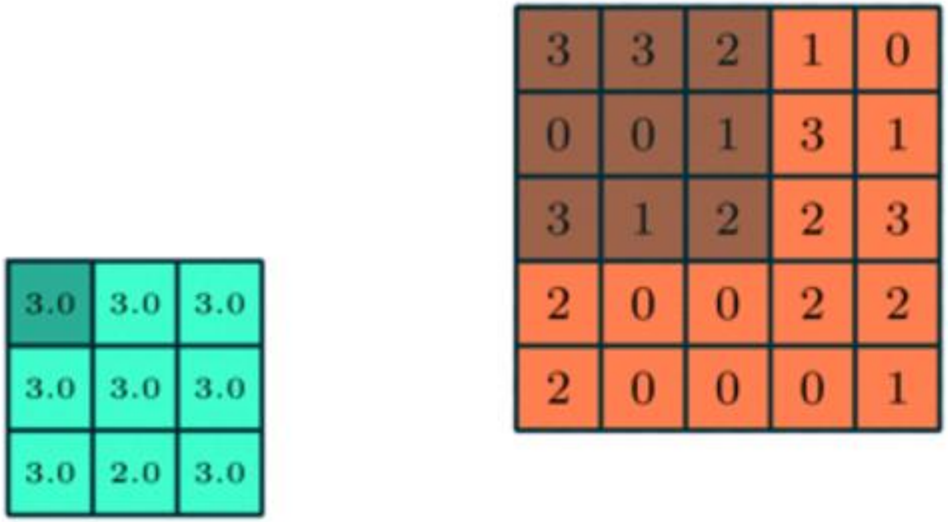
- Pooling

3.0	3.0	3.0
3.0	3.0	3.0
3.0	2.0	3.0

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

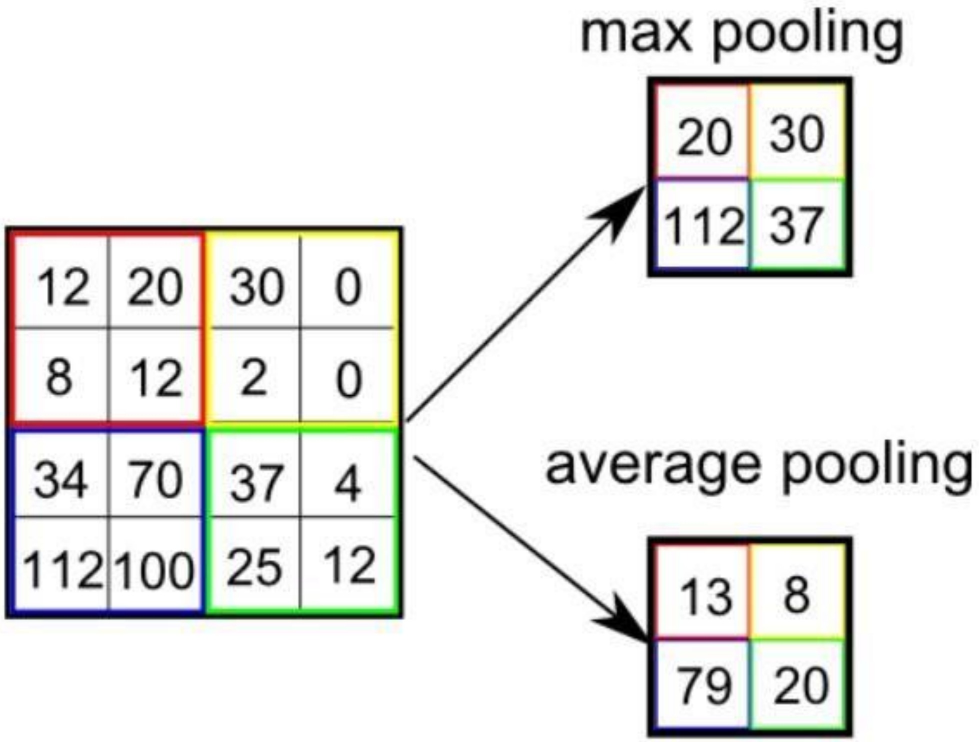
Convolutional neural network

- Pooling

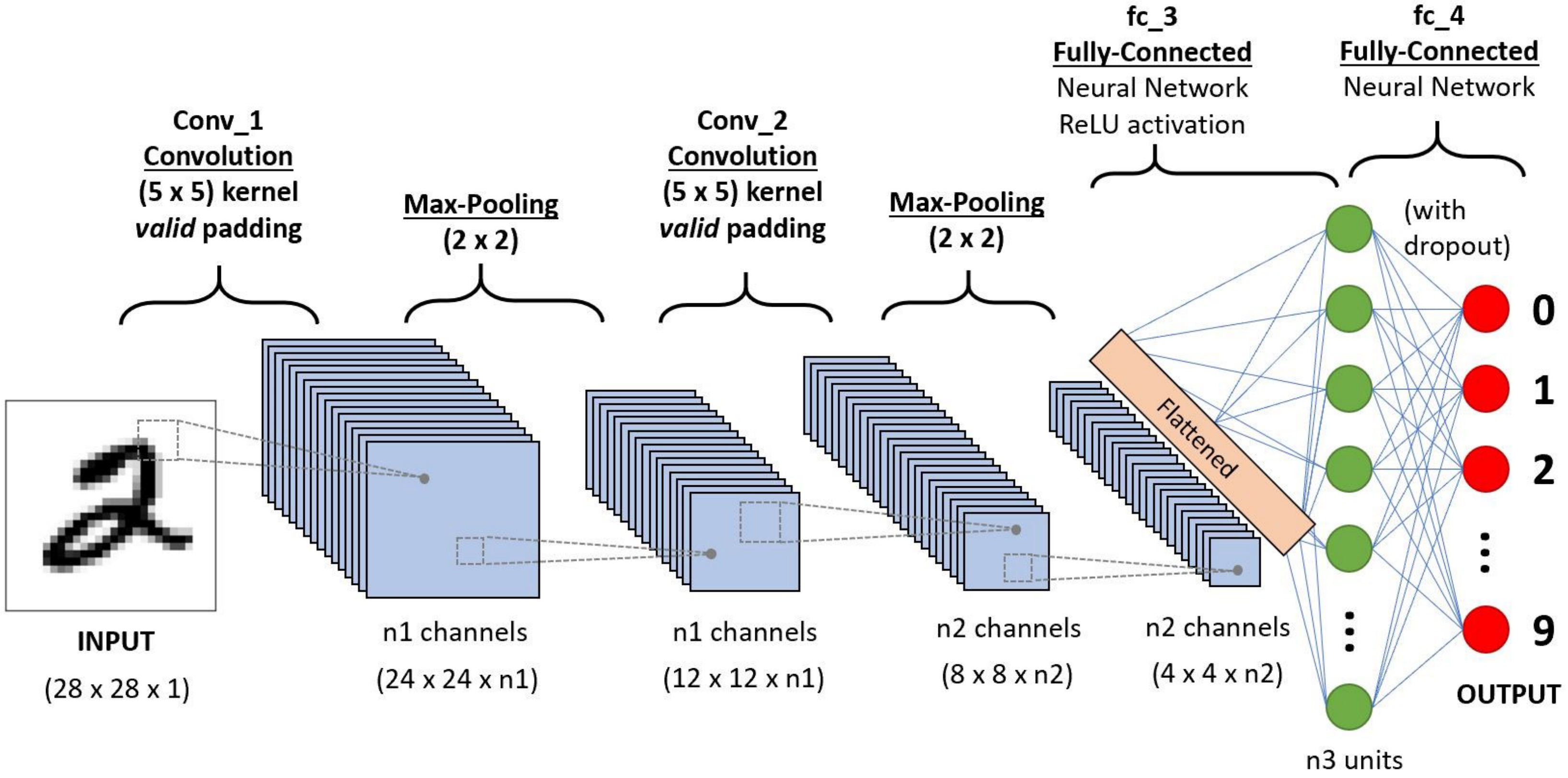


Convolutional neural network

- Max pooling vs Average pooling



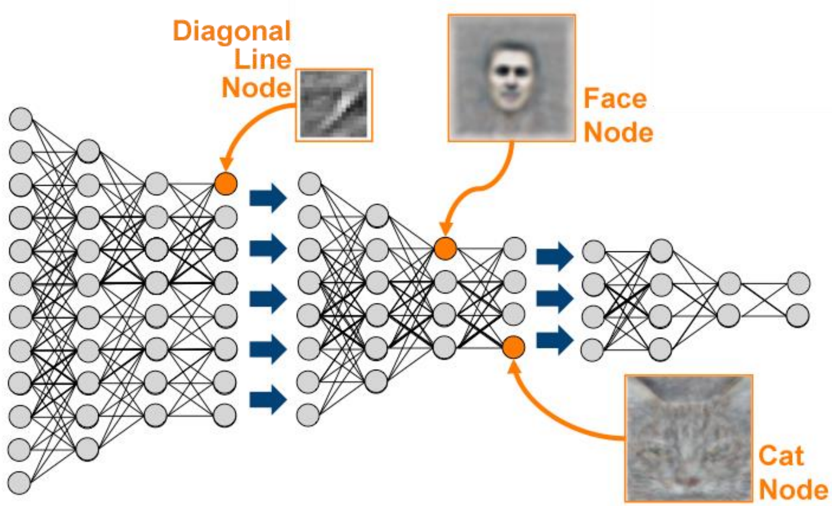
Convolutional neural network



Learn high level features of a cat

"Best neuron" activation heat map

- Training: 16.000 CPU during 3 days
- Learned high levels features of cats, human faces by watching Youtube videos
- Totally unsupervised : unlabeled data



High level feature learning

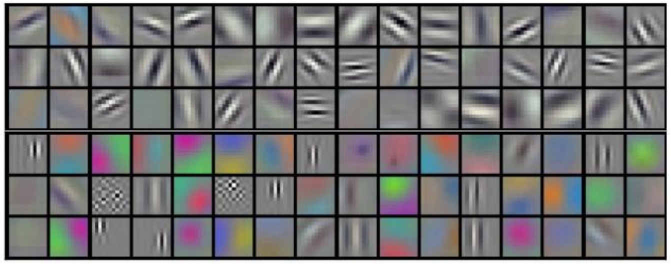
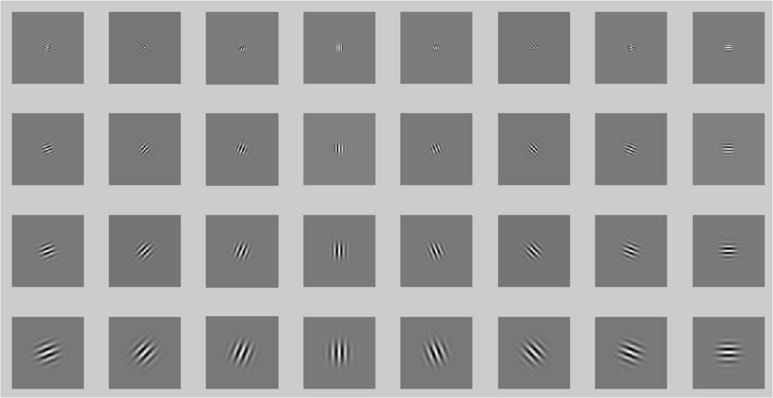


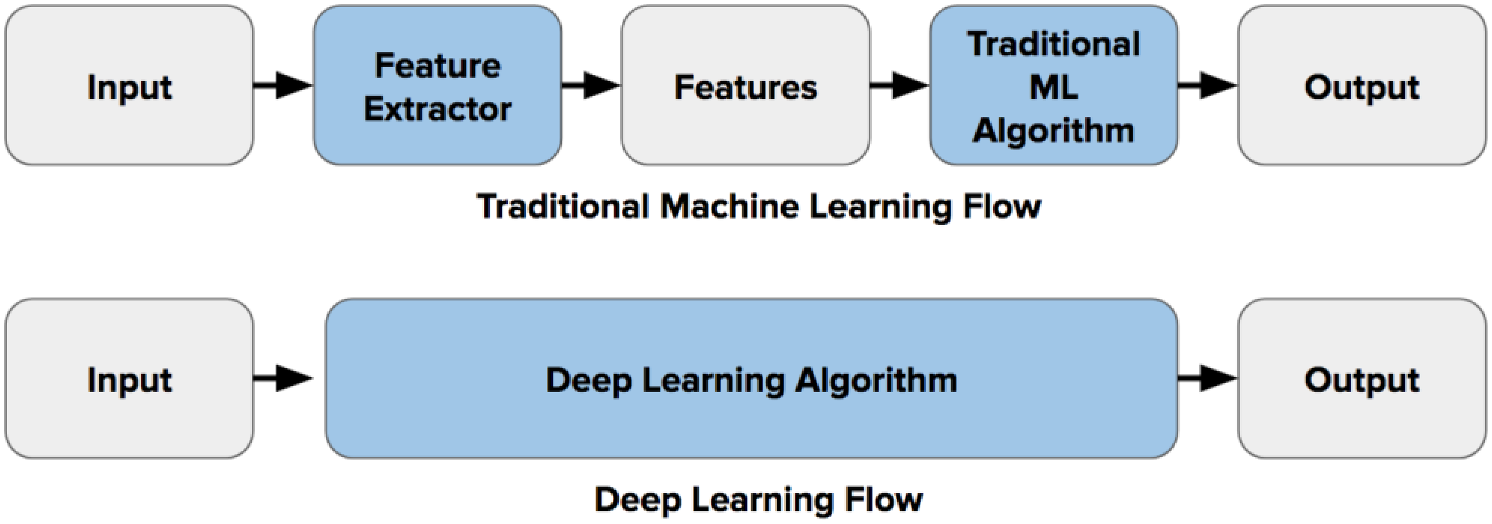
Figure 3: 96 convolutional kernels of size 11×11×3 learned by the first convolutional layer on the 224×224×3 input images. The top 48 kernels were learned on GPU 1 while the bottom 48 kernels were learned on GPU 2. See Section 6.1 for details.

The model learns some edge detection filter. We find a similar process in the cells of the primary visual cortex of the human brain

Edge detectors filters :



Traditional ML vs Deep learning



Practice with Google Colab

https://colab.research.google.com/drive/1UA-wUXx2QziKgZXTDLVo1gn0xImlwj2H

Model evaluation

- Measurements
  - Precision, recall
  - Accuracy
  - F1-score
- Depending on problems, we need some suitable measures

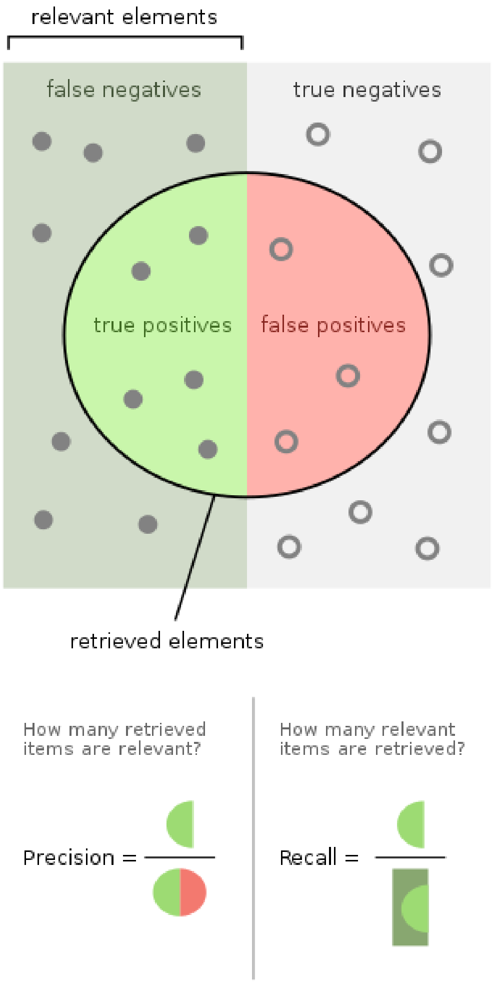
Model evaluation

- $\text{Precision} = \dfrac{\text{True positive}}{\text{Predicted positive}}$

- $\text{Recall} = \dfrac{\text{True positive}}{\text{Positive}}$

- $\text{F1} - \text{score} = \dfrac{2 \text{ x Precision x Recall}}{\text{Precision} + \text{Recall}}$

- $\text{Accuracy} = \dfrac{\text{True positive} + \text{True negative}}{\text{Positive} + \text{Negative}}$



Source: Wikipedia

Confusion matrix

		Predicted				Total
		Cat	Dog	Tiger	Wolf	
Actual	Cat	**6**	0	3	1	10
	Dog	2	**4**	0	4	10
	Tiger	3	3	**3**	0	9
	Wolf	1	4	1	**2**	8
Total		12	11	7	7	

```
>>> sklearn.metrics.classification_report
>>> y_true =  [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …]
>>> y_pred = [0, 0, 0, 0, 0, 0, 2, 2, 2, 3, …]
>>> target_names = ['Cat', 'Dog', 'Tiger', 'Wolf']
```

```
              precision    recall  f1-score   support

         Cat      0.500     0.600     0.545        10
         Dog      0.364     0.400     0.381        10
       Tiger      0.429     0.333     0.375         9
        Wolf      0.286     0.250     0.267         8

    accuracy                          0.405        37
   macro avg      0.394     0.396     0.392        37
weighted avg      0.399     0.405     0.399        37
```

Thank you for your attention