

# Deep Learning et Réseaux de Neurones

Marie Beurton-Aimar, LE Van Linh

September 23, 2025

# Convolutional Neural Network

## AlexNet (from Alex Krizhevsky)

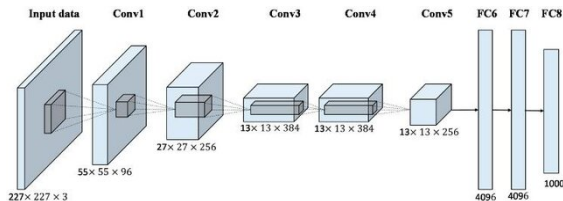
- To classify ImageNet items.
- Won the ILSVRC challenge with an error rate of 15.3 %



# Convolutional Neural Network

## Layer size

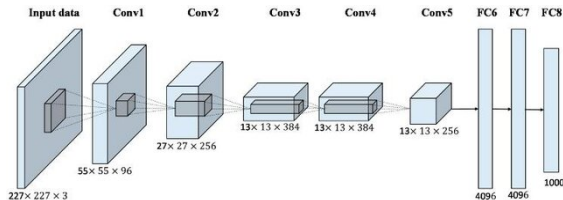
- How to adjust each layer size from input?



# Convolutional Neural Network

## Layer size

- How to adjust each layer size from input?
- Equation to respect :  $(W - F + 2P)/S + 1$   
where  $W$  is the input size,  $P$  is the zero padding and  $F$  the convolutional filter size.  $S$  is the value of stride.
- From the first layer to the second one :  $(227 - 11)/4 + 1 = 55$   
stride=4, no padding and filter = 11



# Convolutional Neural Network

## ReLU function

- The activation function suppressing the neagative value : :  $F(x) = \max(0, x)$
- May be a problem and block the gradient backpropagation.
- Leaky Relu :  $F(x) = \max(0.1x, x)$ . Keep active all neurons.
- Most often chosen in segmentation task.
- Several versions of parametric ReLU exist with an addition of a parameter to change the threshold of activation.
- Gradients could disappear after several layer with more and more small size.
- Replace the tanh function used in LeNet.

# AlexNet vs LeNet

## Comparison: AlexNet vs. LeNet

Feature	LeNet	AlexNet
Input Size	32x32 grayscale	224x224 RGB
Number of Layers	7	8
Dataset	MNIST (digit recognition)	ImageNet (1,000 classes)
Activation	Tanh	ReLU
Regularization	None	Dropout
Pooling	Average Pooling	Max Pooling
GPU Utilization	No	Yes (2 GPUs in parallel)
Parameter Count	~60,000	~60 million

# Convolutional Neural Network

## Dropout Layer

- To prevent overfitting during training.
- It is a regularization technique.
- Randomly remove a part of neurons, typically between 20% and 50% in each layer.
- Each neuron has a chance to be temporarily “dropped”, i.e. set to zero during a particular step.
- Force the network to rely on multiple neurons to represent features at training step.
- Increase the robustness : at the test step, all neurons are used but their weights are scaled down to match the effective capacity of training.
- Can be added after convolutional layers or full connected layers.

# Convolutional Neural Network

## BatchNorm

- Goal : modifying the tensor data in order to obtain an average value equal to 0 and a variance equal to 1.
- Allow to obtain data in the same range than the activation function which are often between 0 and 1.
- As each chanel has the same distribution, they can be mixed and updated in the same way with the same learning rate.
- It is called batch normalisation because this normalisation is done for each batch.



# Convolutional Neural Network

## Data Augmentation

- More the network is huge more data you need.
- To reduce overfitting, modifying data to generate new samples is the usual way to do.
- Operations available : random cropping, horizontal flipping, angle rotation, modifying color channel, filter addition, light changes, creating patches ...



# Convolutional Neural Network

## Synthetic Data

- GAN : Generative Adversarial Networks

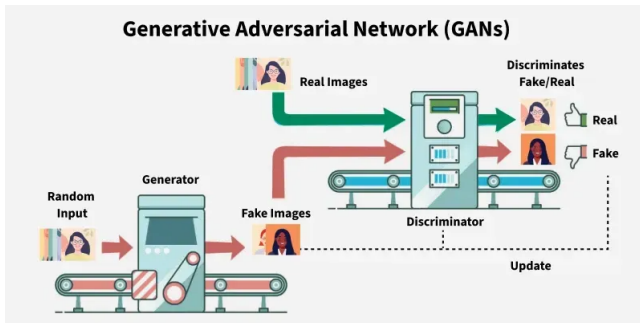


Figure: from <https://www.geeksforgeeks.org/deep-learning/generative-adversarial-network-gan/>

# Convolutional Neural Network

## VGG - the next one

- After AlexNet victory, new architectures have been proposed for the ILSVRC challenge.
- In 2014 - from Simonyan and Zisserman, Visual Geometry Group : VGG16 and VGG19.
- Main idea :
  - reduce the convolution kernel to 3X3.
  - add more convolution layers.

# VGG Architecture

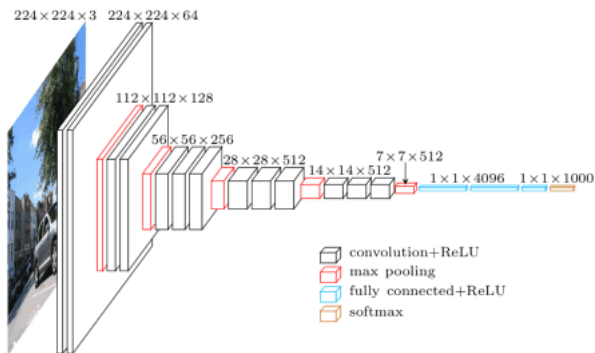
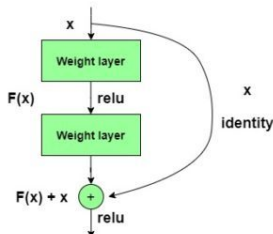


Figure: from <https://viso.ai/deep-learning/vgg-very-deep-convolutional-networks/>

# ResNet Architecture

## What is ResNet?

- From Kaiming He et al. in 2015 “Deep Residual Learning for Image Recognition” (computer vision research paper).
- Problem to solve : when the number of layers increases the accuracy level can get saturated, not by overfitting but vanishing or exploding gradient values.
- Introducing the skip connections to set up shortcuts for the gradient to pass through.  $y = F(x) + x$



# ResNet Architecture

## Residual Block

- Each residual block consists of:
  - A 3x3 convolution layer followed by a batch normalisation layer and a ReLU activation.
  - Another 3x3 convolution layer and a batch normalisation layer.
  - The skip connection skips both convolution layers and adds the inputs directly before the ReLU activation function.

# ResNet Architecture

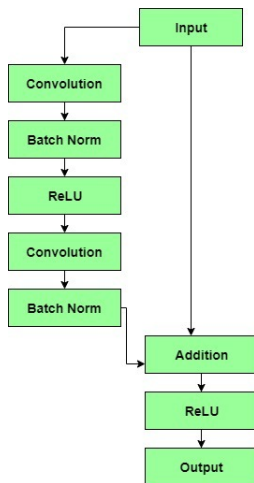


Figure: from <https://www.geeksforgeeks.org/machine-learning/introduction-to-residual-networks/>

# ResNet Architecture

## Usage

- ResNet is usually used as a main backbone in a lot of architectures.
- ResNet50 is the most popular.
- Mainly used for all image processing : image classification, semantic segmentation, object detection ...
- The backbone role is to extract features from the images and to produce feature maps at different levels of resolution.
- The feature maps at the lower level contain spatial information.
- The feature maps at the higher level have richer semantic information due to their large receptive field.



# Feature Maps

## Description

- A way to capture the output activations of convolution layers.
- Provide insights into how the network processes and interprets input data at various stages.
- Feature maps are outputs of filters or kernels applied to input image by convolution layers.
- Able to capture patterns present in the input image.
- In PyTorch it is possible to visualize with the help of torchvision.

# Feature Maps



Figure: from <https://www.geeksforgeeks.org/deep-learning/visualizing-feature-maps-using-pytorch/>

# Feature Maps

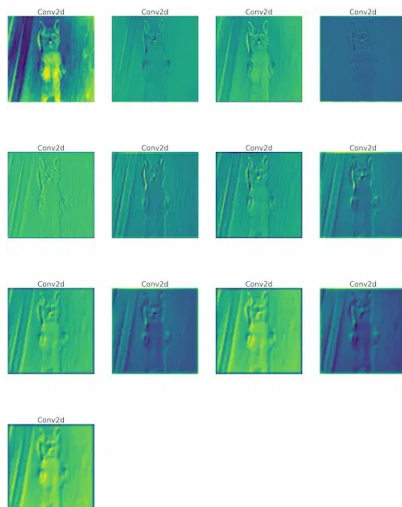


Figure: from <https://www.geeksforgeeks.org/deep-learning/visualizing-feature-maps-using-pytorch/>