23/5/2020

CNN

① What are CNNs?

- ↳ video starts with showing ambiguous features
- ↳ Brain analyzes certain features and classifies according to those features.
- ↳ Pertaining to image processing.

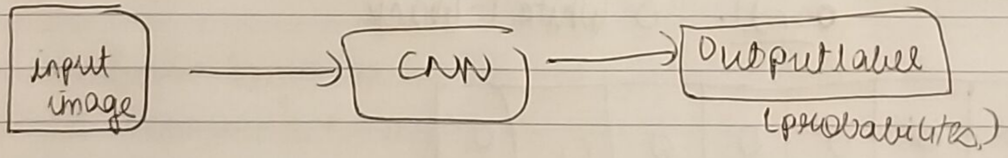
idea

→ algorithm makes guesses and provides probabilites of the most likely answers. based on features

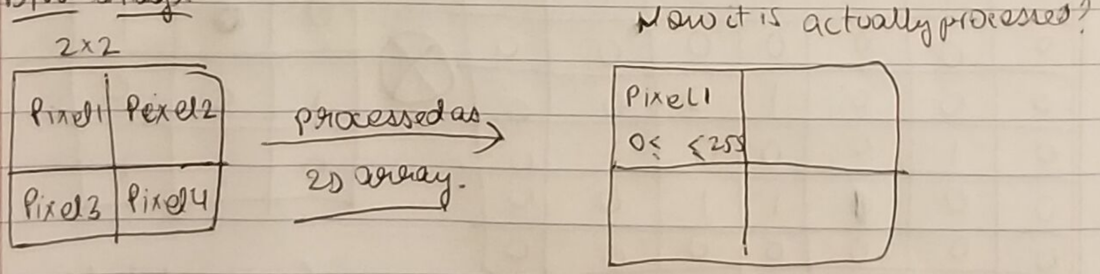
→ ex. Self driving cars, image tagging on fb.
↳ recognize people, obstacles

Yann Lacun - prominent in CNN.

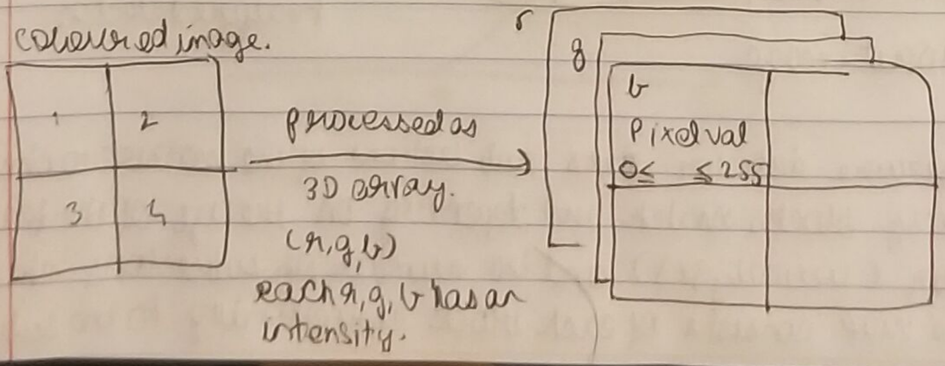
Main structure



B/W image



coloured image.



Step1 : Convolution

1(b) ReLU Layer.

Step2 : Max pooling

Step3 : Flattening

Step4 : Full connection

② Step1 : convolution - goal - find features and maintain spatial relations between features.

$$(f * g)(t) \overset{def}{=} \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau$$

=> Feature detector / Kernel / filter

Assumption - Dealing with B/W image with pixel value ranging b/w 0 and 1. 0 - White 1 - black

0	0	0	0	0	0	0
0	1	0	0	0	1	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	1	0	0	0	1	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

Input image

$\otimes$

0	0	1
1	0	0
0	1	1

Feature detector

=

0	1	0	0	0
0	1	1	1	0
1	0	1	2	1
1	4	2	1	0
0	0	1	2	1

Feature map

-> The feature detector scans sub sections of the matrice to check how many blocks match and depending on the size of the feature detector (usually 3x3) and the number of subsections checked a feature's map consists of each block corresponding to the subsections with a number as to how many matched between subsection and feature ~~map~~ detector.

→ The sequence in which subsections are checked are called Strides. → determines ~~how~~ how many pixels to skip

→ usually a stride of 2

→ Result → reduced size of image.

→ We are lossing information but the goal of the feature detector is to find integral features of image.

→ We can have multiple feature detectors ⇒ multiple feature maps. ⇒ to detect multiple features.

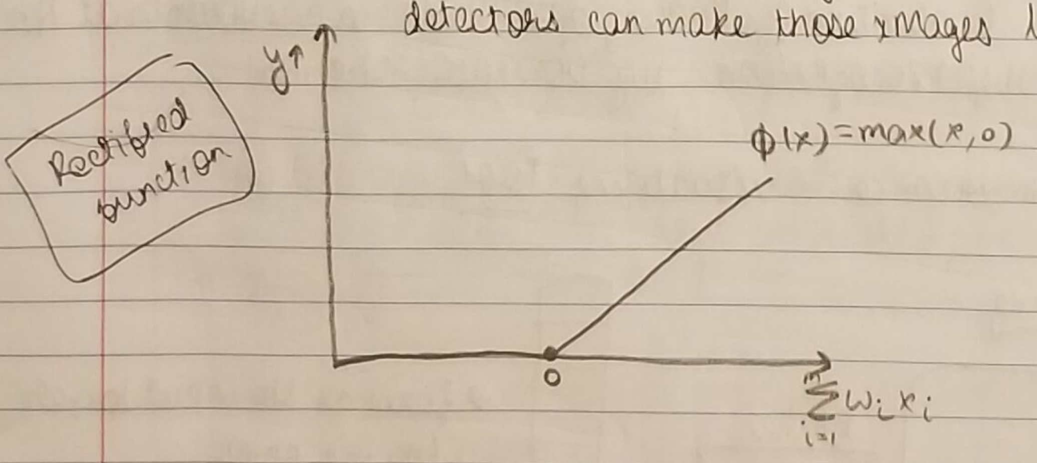
↳ in turn different versions of the input image.

→ examples of features detectors - edge detection, sharpen

③ ReLU Layer. - Rectified Linear Unit.

→ Convolution layer provides the feature maps

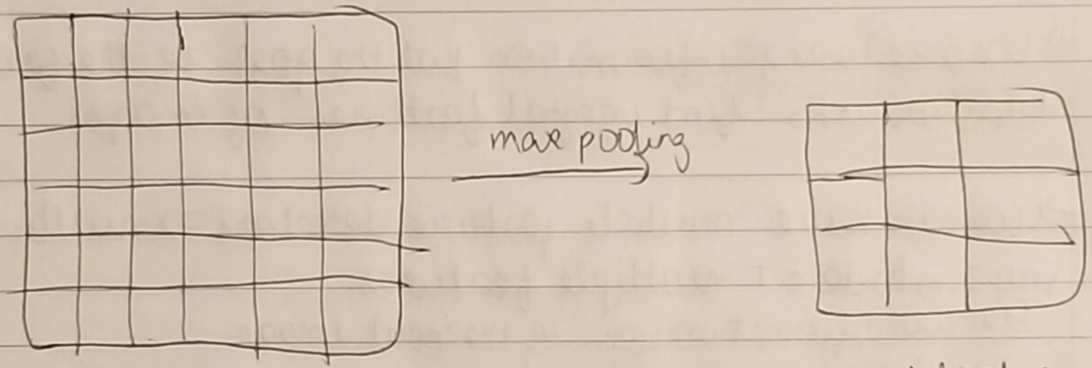
→ Once these are determined, rectifier function is applied to INCREASE NON LINEARITY, elements of images are non linear but feature detectors can make those images linear.



(4) Pooling

Spatial invariance - recognize features in different versions | orientations of an object.

max pooling



feature map.

pooled feature map.

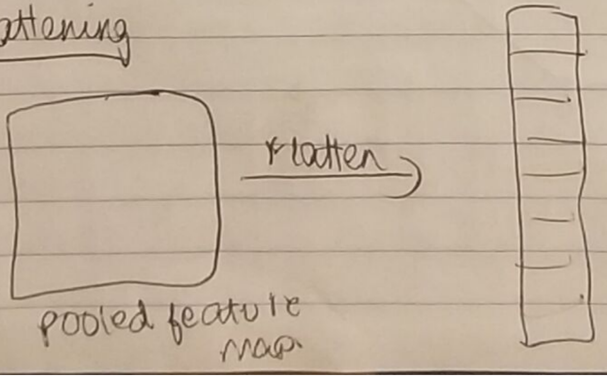
maxpool

- → select max number from sub section of feature map.
- → concept of stride applies here as well.
- → reducing size and preserving important features
- → reducing the numbers of parameter to prevent overfitting.

⇒ paper: Evaluation of Pooling operations in convolutional Architectures for Object Recognition. by Dominic Scherer

⇒ scs.ryerson.ca. - visualizing tool.

(5) Flattening

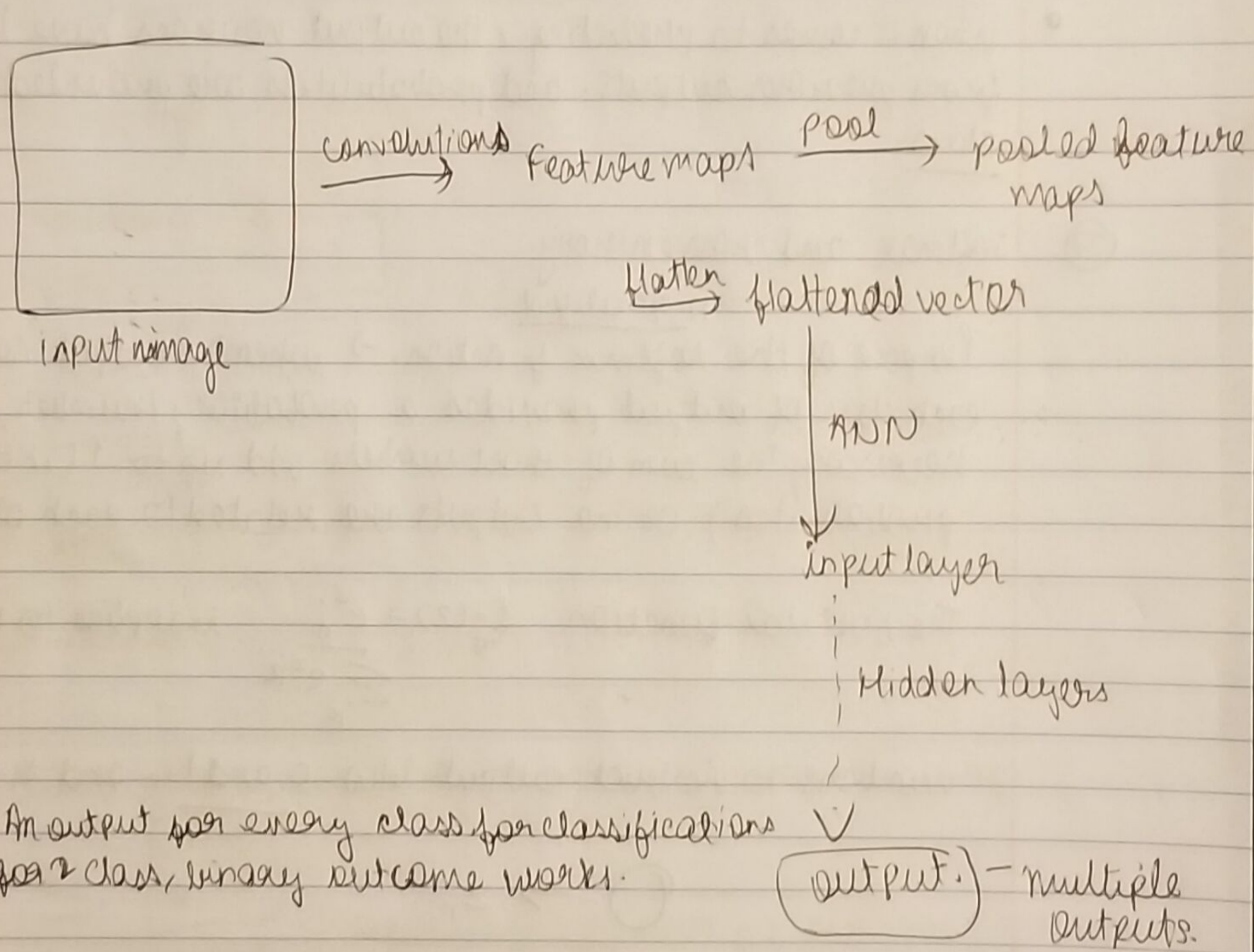


pooled feature map

→ form as the input layer for the ANN.

⑥ Full connection

Add ANN to the CNN



→ An output for every class for classifications
→ for 2 class, binary outcome works.

- The goal is to combine attributes to better predict the class. using the hidden layers. to further optimize.

- concepts of backpropagation apply to minimize loss functions
→ feature detectors are adjusted in the process.
→ weights are adjusted.

→ each class has an output neuron. - the signals and adjusted weights from hidden layer ~~signal~~ send the same signal to output neurons and based on the training the output neuron makes decisions on the signals.

- Not all neurons from the hidden layer will fire at same but overtime output neurons develop associations with certain neurons to make better decisions and learn over iterations.

- when it comes to predicting, the output neurons have learnt from previous outputs. and probabilities are provided for each class.

⑦ Softmax and cross entropy.

normalized

Purpose of the softmax function. → when the output neuron for each class of output provides a probability / number, it is not necessary the sum of those numbers add up to 1 (in the case of probabilites) or the outputs are related to each other.

The softmax function $f_j(z) = \frac{e^{z_j}}{\sum_k e^{z_k}}$ is applies to those numbers to restrict output b/w 0 and 1 and the sum = 1

○ → z_1

○ → z_2

Cross entropy function:

$$H(p,q) = -\sum_x p(x) \log q(x)$$

similar to mean squared error function and is used to assess values provided by the softmax to assess the CNN.

example for evaluating

			NN1	NN2
Dog pic	Dog	1	0.9	0.6
	cat	0	0.1	0.4
cat pic	Dog	0	0.1	0.3
	cat	1	0.9	0.7
Dog pic	Dog	1	0.4	0.1
	cat	0	0.6	0.9

① Classification Error - Right/wrong - 1/3 - How many wrong

	NN1	NN2
② MSE	0.25	0.71

↳ Better performance

	NN1	NN2
③ Crossentropy -	0.38	1.06

} NN1 - better performance.

Advantage of cross entropy - Better in assessing even when output value of the model are very low compared to the final value in the initial steps of backpropagation.