DL L12 : 30.05.2022

4. depth shortening

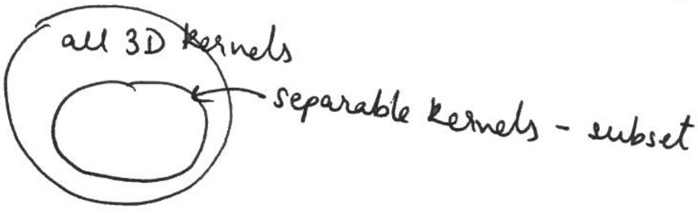
Reduce no of i/p channels

Slide 7-12
13

2) - 4) are approximations of 1)

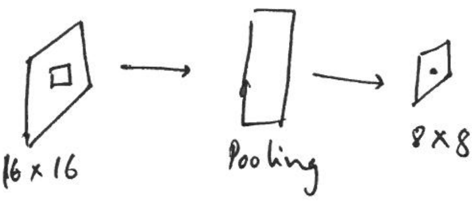
+ reduced complexity

- mostly at the price of a performance degradation



7.3 Pooling and unpooling layer

2D Pooling layer with stride $s \in \mathbb{N}$



eg

downsampling

- max pooling : P = max of a, b, c....
- mean pooling : P = mean "
- L2-norm pooling : P = L2 norm "

Mostly we use the max pooling

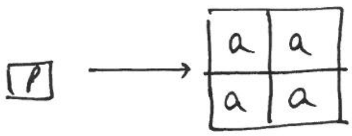
Slide 7-14
15

Q Difference b/w translation-invariant vs Translation ~~equivalent~~ equivariant.

no change in translation

if the ~~image~~ signal is shifted by 1 time frame, the o/p is also shifted by 1 time frame.

2D unpooling layer eg 2 × 2



upsampling to restore the original image size after certain pooling

7.4 Deconvolutional layer

Slide 7-16

deconvolutional layer ≠ deconvolution
≙ fractionally strided convolution
≙ transposed strided convolution
≙ learnable upsampling

(the first three lines above: } bad names)

Goal
- replaces unpooling
- increases spatial size/resolution without changing the signal shape in the input

Slide 7-17 → lowpass filter with a fixed kernel

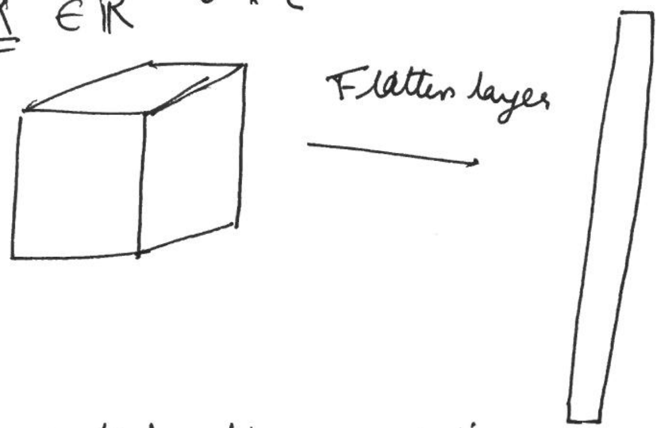
deconvolutional layer =

- zero insertion
- convolution (low pass filtering) with a <u>learnable</u> kernel

Slide 7-18

<u>7.5</u> <u>Flatten layer</u>

$\underline{\underline{X}} \in \mathbb{R}^{H \times W \times C}$ $\xrightarrow{\text{Flatten layer}}$ $vec(\underline{\underline{X}}) \in \mathbb{R}^{HWC \times 1}$



as an interface b/w convolutional layer and dense layers.

conv layers : feature learning → flatten layer → dense layer (combine and interpret the features) → output layer (make a decision)

Flatten layer can increase the number of computational complexity for the later dense layer as its weight matrix is dependent on N^2 i.e. quadratic on previous layer.

7.6 Global average pooling layer

GAP: more efficient attention layer than flatten layer.

$$\underline{\underline{X}} = [x_{hwc}] \in \mathbb{R}^{H \times W \times C} \xrightarrow[GAP]{} \left[\frac{1}{HW} \sum_h \sum_w x_{hwc}\right] \in \mathbb{R}^C_{1 \le c \le C}$$

- replace each feature map by its average
- much less neurons and weights in the final dense layers
- suitable for
  - ▵) large DNN : reduced complexity
  - ▵) small dataset : reduces overfitting

7.7 Architecture of CNNs

A CNN consists of mainly

- convolutional layer – for heirarchical features learning
- pooling layer – to get abstract features

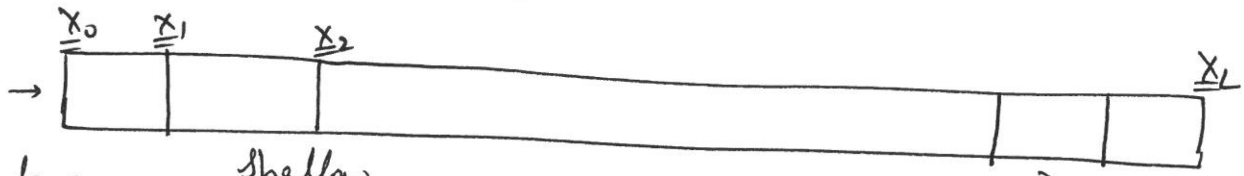
Optionally

- unpooling or deconvolutional layer
- flatten or GAP layer
- dense layers

Why deep CNN?

1. Need deep architecture to extract heirarchical features from input

2. large model capacity for difficult tasks.



layers   Shallow   Deep

features   low level details eg edges, orientations, corners, etc   high level abstract features eg class, semantic meaning, etc.

Slide 7-19