

A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines. Some nodes are highlighted with blue circles or dots, while others are grey. The lines are thin and grey, creating a mesh-like structure.

CNN Tutorial

정승환

2018.07.07

A decorative network diagram in the bottom-right corner, similar to the one in the top-left. It shows a network of nodes and lines, with some nodes highlighted in blue and others in grey.

About Me

2015.09 ~ current	Machine Learning cell, SK Telecom
2013.08 ~ 2015.08	Consulting DA, Deliotte Consulting
2011.07 ~ 2013.07	Entrue Consulting, LG CNS
2011.02 ~ 2011.06	컨버전스 지식센터, 한국 생산성 본부(KPC)
2009.02 ~ 2011.02	M.A in Industrial Engineering, KAIST
2002.02 ~ 2009.01	B.A in Industrial Engineering, Hanyang Univ.

About Me

- 2015.09 ~ current Data 분석 기반 Targeting 고도화 방안
AutoML 알고리즘 수립 및 운영
Structure Data 기준 Deep learning 적용 및 운영
- 2014.03 ~ 2014.09 SSG.com 애널리틱스 프로젝트
- 진성고객 산출
 - 가격 정책 고도화 방안
- 2013.11 ~ 2014.11 KBO 프로야구 일정 수립(2014년, 2015년 적용)
- 2012.04 ~ 2013.06 LGD 분석 CoE(Center of Excellence) 와 분석 PJT 수행
- Early Warning 고도화 시스템 제안
 - IT 모바일 수요 예측
 - SPM(Service Part Management) 적확도 향상 방안

CNN 주요 특징

- 각 레이어의 입출력 데이터의 형상 유지
- 이미지의 공간 정보를 유지하면서 인접 이미지와의 특징을 효과적으로 인식
- 복수의 필터로 이미지의 특징 추출 및 학습
- 추출한 이미지의 특징을 모으고 강화하는 Pooling 레이어
- 필터를 공유 파라미터로 사용하기 때문에, 일반 인공 신경망과 비교하여 학습 파라미터가 매우 적음

CNN 주요 개념

- Convolution
- Channel
- Filter(=Kernel)
- Stride
- Padding



Convolution

- 연속 변수 -

$$y(t) = (x * w)(t) \\ = \int_{-\infty}^{\infty} x(a)w(t-a)da$$

- 이산 변수 -

$$y[n] = \sum_{a=-\infty}^{\infty} x[a]w[n-a]$$

Convolution

Convolution?

- 이산 변수 -

w 함수를 n만큼 평행이동 시킨 함수와 x 함수와의 곱한 값을 x 축에 대해 전부 더한 값들의 집합

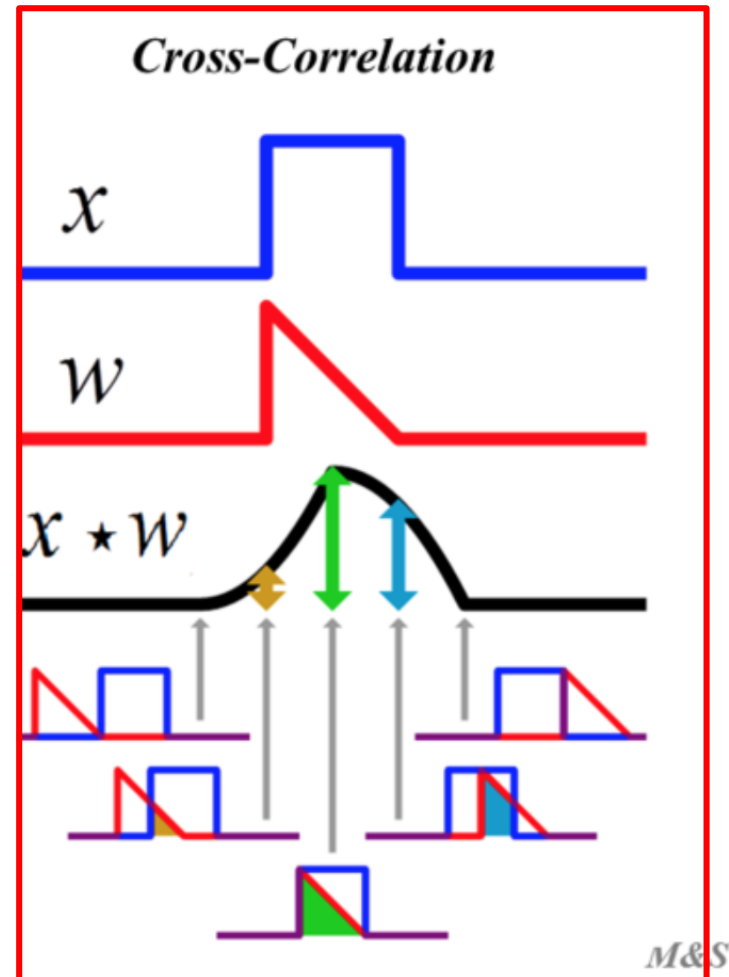
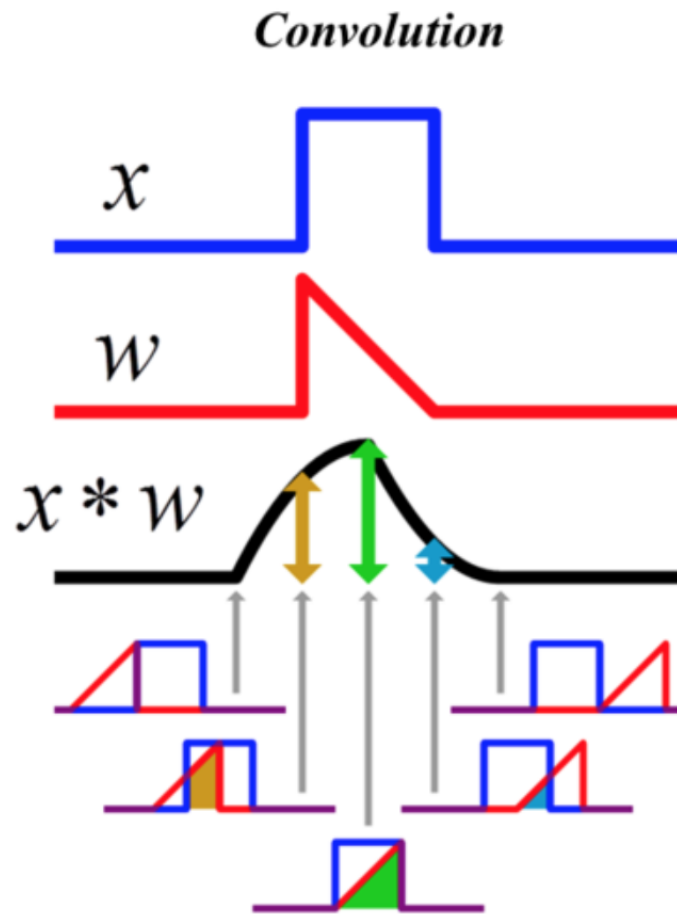
$$y[n] = \sum_{a=-\infty}^{\infty} x[a]w[n-a]$$

$$x[a]w[a] \xrightarrow{\text{y축 대칭}} x[a]w[-a]$$

$$x[a]w[-(a-n)]$$

n만큼 평행이동

Convolution



Convolution

$$y[i, j] = (x * w)[i, j]$$

$$= \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} x[m-i, n-j] w[m, n]$$

1 _{x1}	1 _{x0}	1 _{x1}	0	0
0 _{x0}	1 _{x1}	1 _{x0}	1	0
0 _{x1}	0 _{x0}	1 _{x1}	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

Convolved
Feature

Channel

RED Channel



Green Channel



Blue Channel



Filters

2	2	4	4	0
80	60	10	7	10
4	10	80	10	20
12	24	10	8	20
22	42	20	10	10

Images

\times

	0	0.2	0	
	0.2	0.4	0.2	
	0	0.2	0	

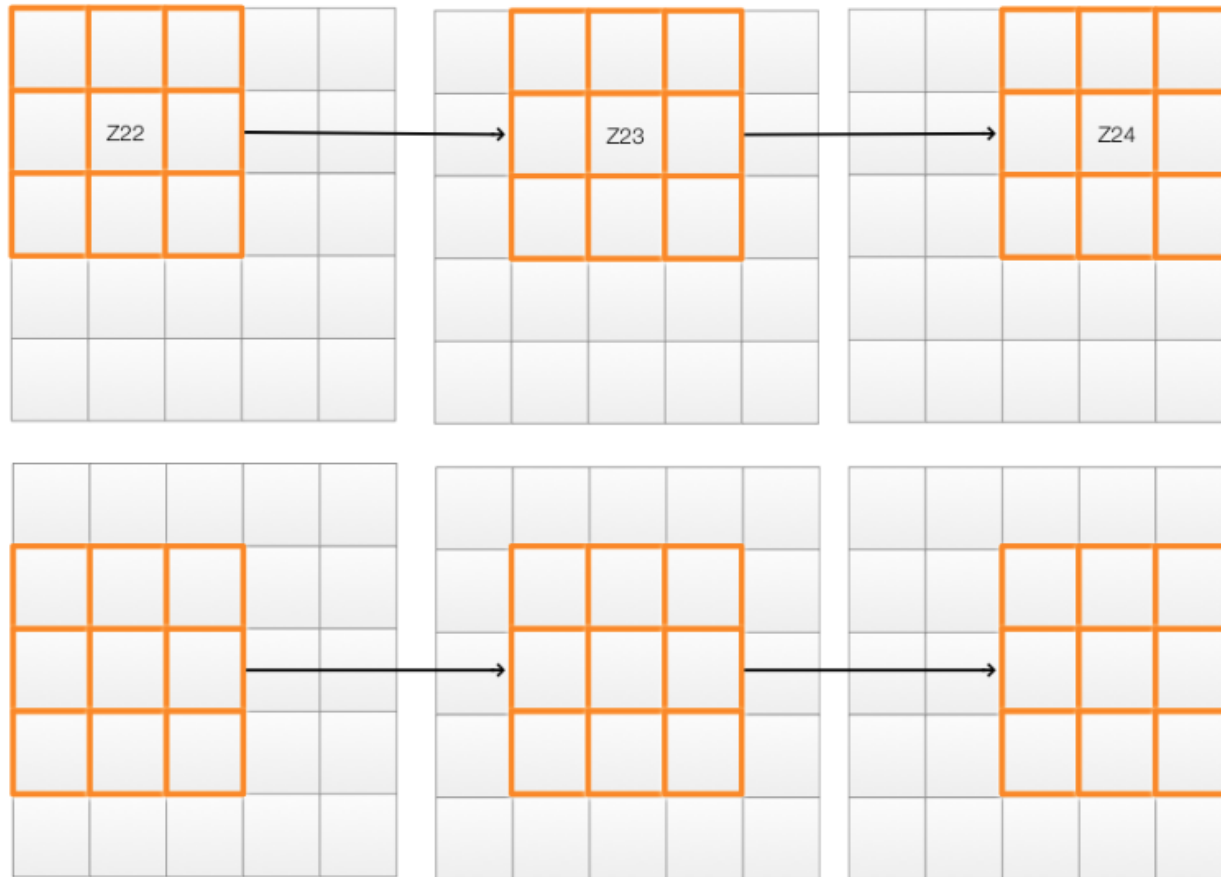
3 * 3 filters

=

		40		

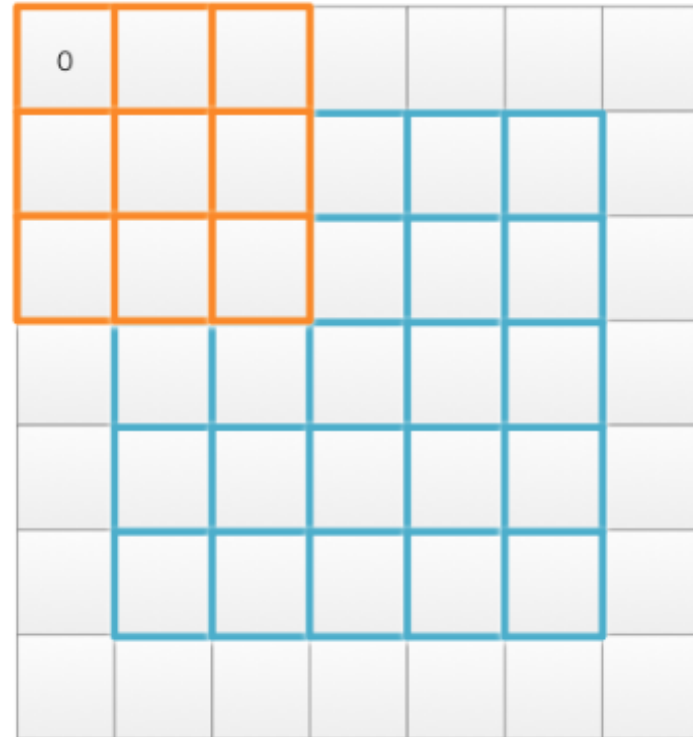
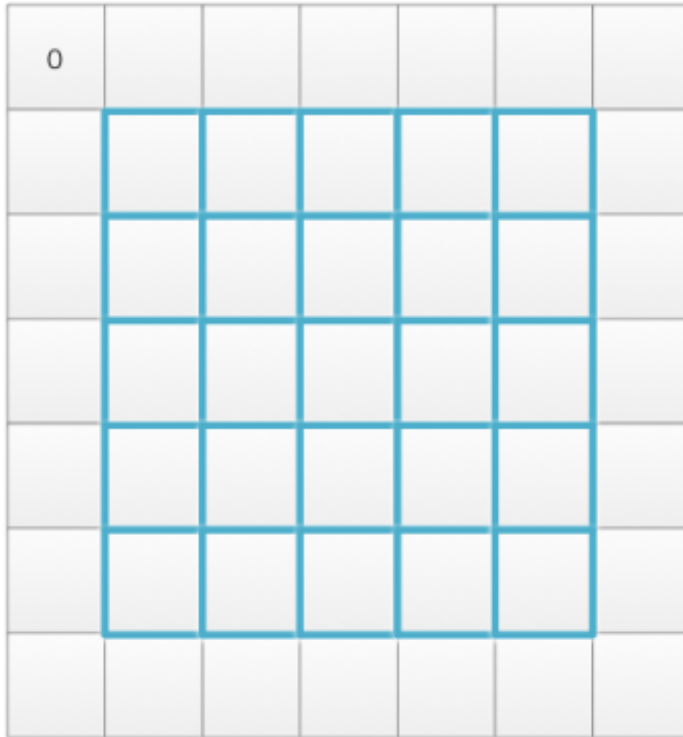
$$80 \times 0.4 + 4 \times 0.2 \times 10 = 40$$

Strides



stride = 1

Padding



Pooling

Activation Map

12	20	30	0
8	12	2	0
34	70	37	7
112	100	22	12

Max Pooling

20	30
112	37

Average Pooling

13	8
79	18

CNN Process



CNN Output size

- 입력 데이터 높이: H
- 입력 데이터 폭: W
- 필터 높이: FH
- 필터 폭: FW
- Strid 크기: S
- 패딩 사이즈: P

$$OutputHeight = OH = \frac{(H + 2P - FH)}{S} + 1$$

$$OutputWeight = OW = \frac{(W + 2P - FW)}{S} + 1$$

CNN Output size

$$OutputHeight = OH = \frac{(H + 2P - FH)}{S} + 1$$

$$OutputWeight = OW = \frac{(W + 2P - FW)}{S} + 1$$

Example

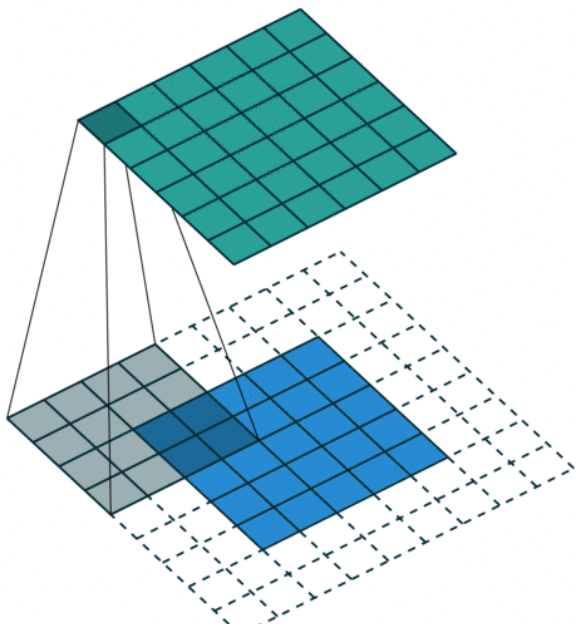
- 32 X 32 이미지에 kernel_size = 3, stride=1 padding =1 인 경우

$$= (32 + 2*1 - 3) / 1 + 1 = 32$$

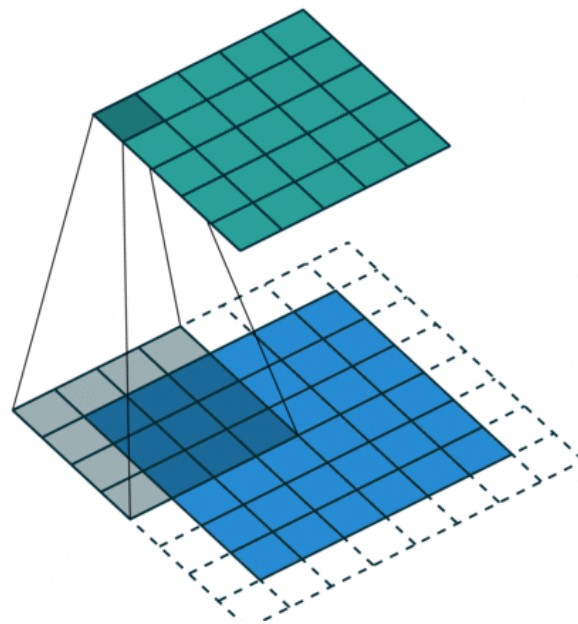
Transposed CNN

CNN을 역으로 적용함

- 일반적으로 CNN은 이미지의 size를 줄여가는 형태로 진행
- input data 보다 이미지의 크기를 크게 할 때 적용



CNN



Transposed CNN



Thanks!

Any questions?

