

# CNN 개념 복습 - 인혜

왜 cnn을 쓰는게 이미지 처리에서 좋은지

cnn이 이미지를 인식하는 방법 - conv layer

filter의 역할과 feature map

padding

왜 padding이 필요한가?

1. 이미지 데이터의 축소를 막기 위해.

2. Edge pixel data를 충분히 활용하기 위해.

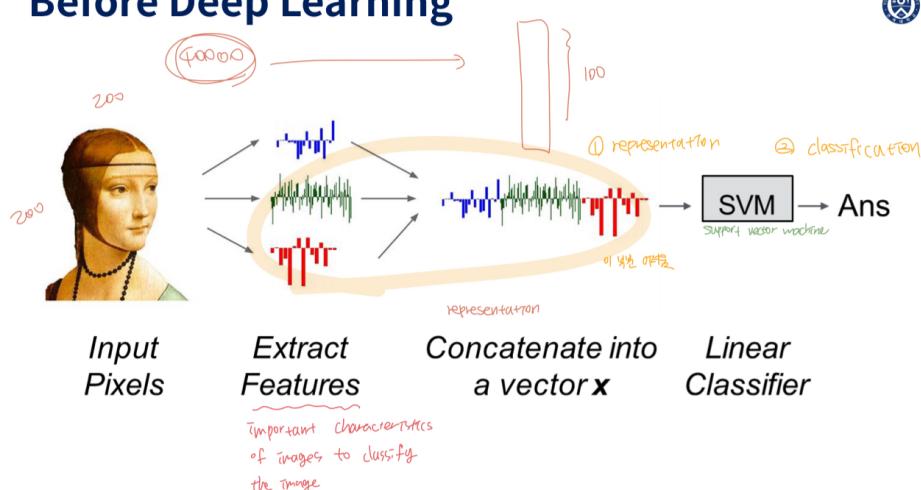
pooling

FC layer

dropout

## 왜 cnn을 쓰는게 이미지 처리에서 좋은지

### Before Deep Learning



연세대학교  
인공지능융합대학

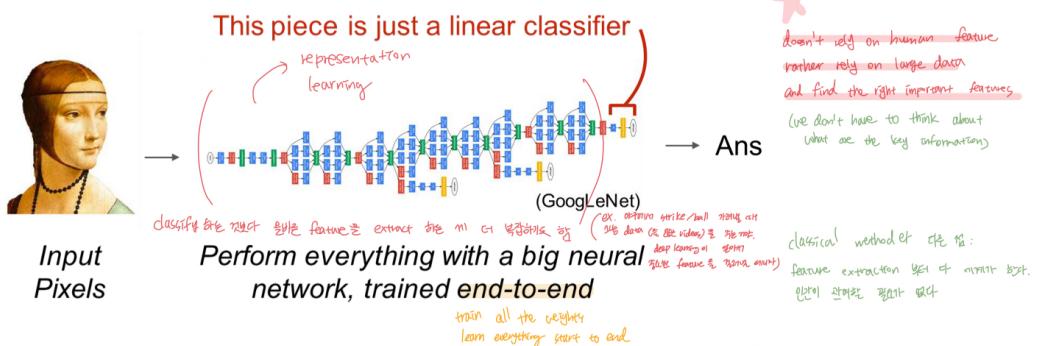


Figure: Karpathy 2016

AIC2120

28

### With Deep Learning



연세대학교  
인공지능융합대학



**Key:** perform enough processing so that by the time you get to the end of the network, the classes are linearly separable

Figure: Karpathy 2016

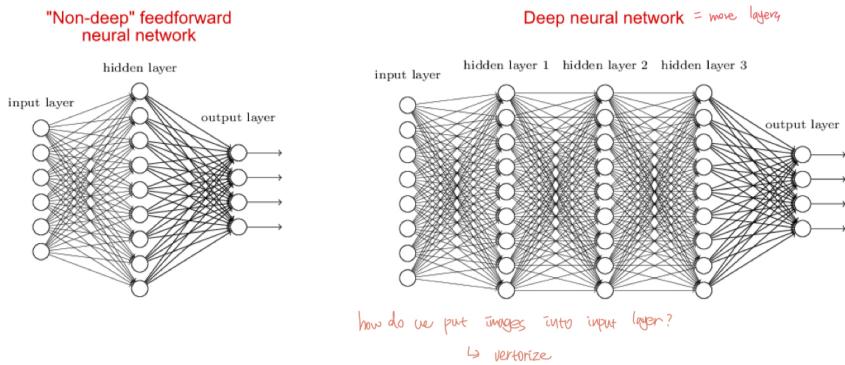
image는 feature extract를 인간이 하기 어려움

따라서 feature extract model 필요

MLP or Fully Connected Layer for Images?

how do we design neural networks??

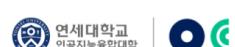
## Multi-Layered Perceptron for Images?



AIC2120

30

## MLP or Fully Connected Layer for Images



Example: 200x200 image

40K hidden units

→ ~2B parameters!!!

too many weights

한번의 neuron의 모든 input pixel을  
접근



- Spatial correlation is local
- Waste of resources + we have not enough training samples anyway..

Ranzato

→ Too many parameters!!

Waste of resources + we have not enough training samples

→ neuron이 image 같지 않음

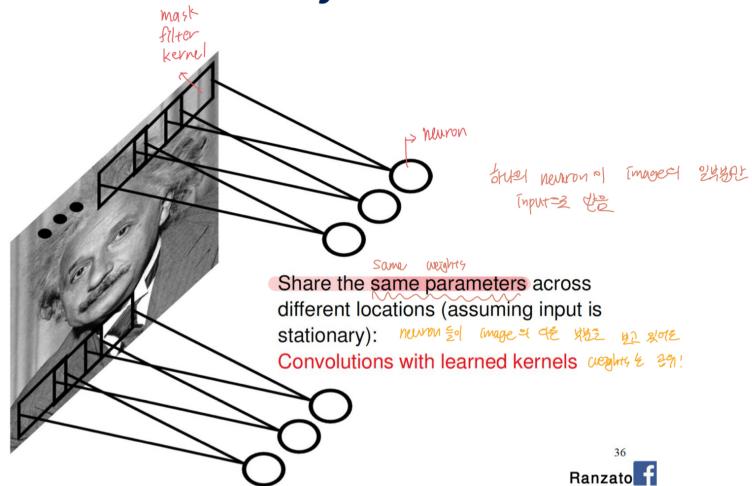
따라서 **효율적인 feature extract**을 위해 CNN을 사용한다.

그렇다면 CNN은 왜 image feature extract에 효율적일까?

## cnn이 이미지를 인식하는 방법 - conv layer

filter(=kernel=mask)의 부분만큼 image를 읽어서 합성곱을 하는 것

### Convolutional Layer

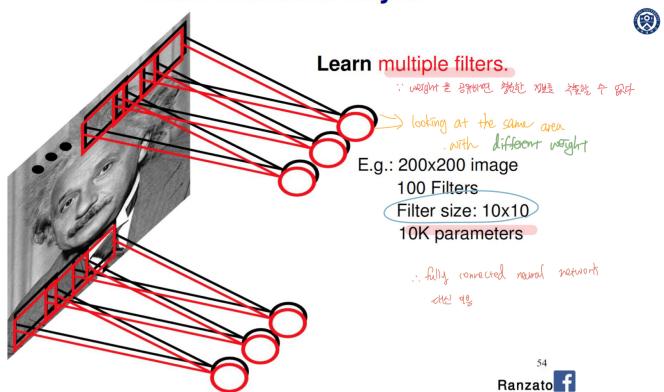


36  
Ranzato f

- 한 번의 conv layer에는 “같은 parameters”를 공유한다.
- conv layer의 과정 시각화 → pdf

## filter의 역할과 feature map

### Convolutional Layer



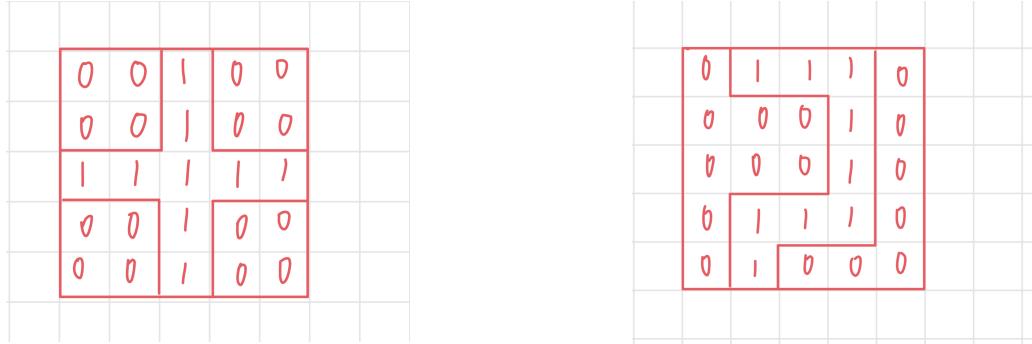
54  
Ranzato f

filter를 적용하는 영역이 filter와 비슷하면 값이 커지고 다르면 값이 작아진다.

ex.

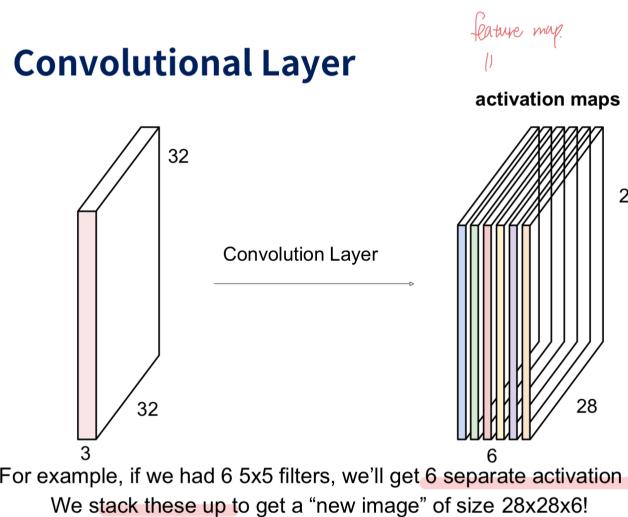
1번 filter

2번 filter

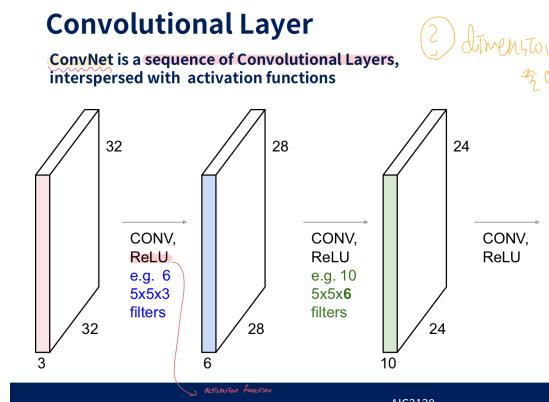


한 가지의 filter를 이용해서 image를 모두 순회하면 neuron의 집합인 feature map이 형성된다.

우리는 multiple filters를 할 예정이므로 여러 개의 feature maps가 stack 된다.



이를 activation function인 ReLU로 보낸다.



## padding

### Zero Padding

zero padding이란 image 주위를 0으로 둘러주는 과정을 말한다.

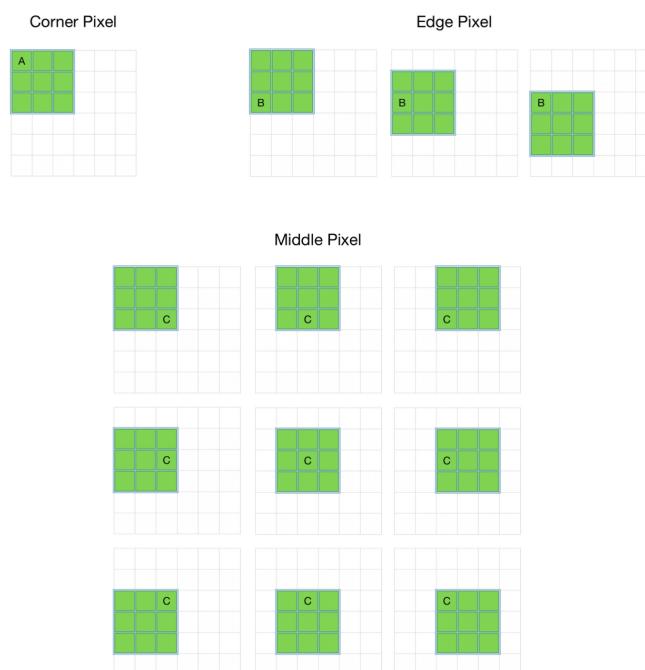
## 왜 padding이 필요한가?

### 1. 이미지 데이터의 축소를 막기 위해.

convolution operation에서 input data인  $(n \times n)$  pixel image가  $(f \times f)$  filter를 통해  $(n - f + 1) \times (n - f + 1)$  pixel image로 축소된다.

CNN 과정에서 위와 같이 여러번의 계산을 해야하는데 초반부터 이미지가 너무 작아져 버리면 더 깊게 학습시킬 데이터가 부족해지는 현상이 만들어진다. 이는 neural network의 성능에 악영향을 미치기 때문에 padding을 이용하여 크기를 조절할 수 있다.

### 2. Edge pixel data를 충분히 활용하기 위해.



#### [CNN] Padding 무엇인가?

Zero Padding - zero padding이란 image 주위를 0으로 둘러주는 과정을 말한다. padding size 딥러닝을 할 때 항상 input과 output의 데이터 사이즈를 잘 알아야 모델을 잘 만들 수 있다.  $p$  : padding layer의 수  $image\_size$  :  $(n \times n)$  padded\_image\_size :  $((n + 2p) \times (n + 2p))$  convolution-operation (with  $(f \times f)$  filter) outputs :  $((n - f + 1) \times (n - f + 1))$

☞ <https://supermemi.tistory.com/entry/CNN-Padding-무엇인가>

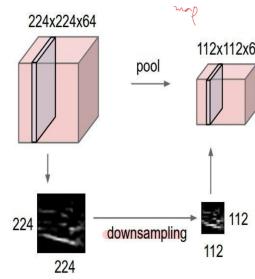
0		0
0		0
0		0
0		0

## pooling

downsampling

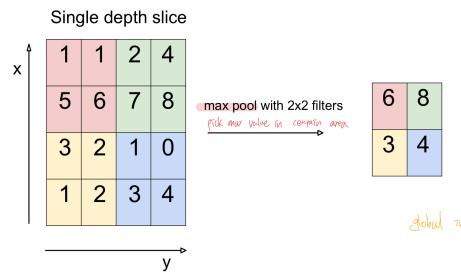
why?

- less computation
- 하나의 neuron이 보고 있는 영역이 전체에서 차지하는 비율이 증가함



max pool

- pick max value in certain area

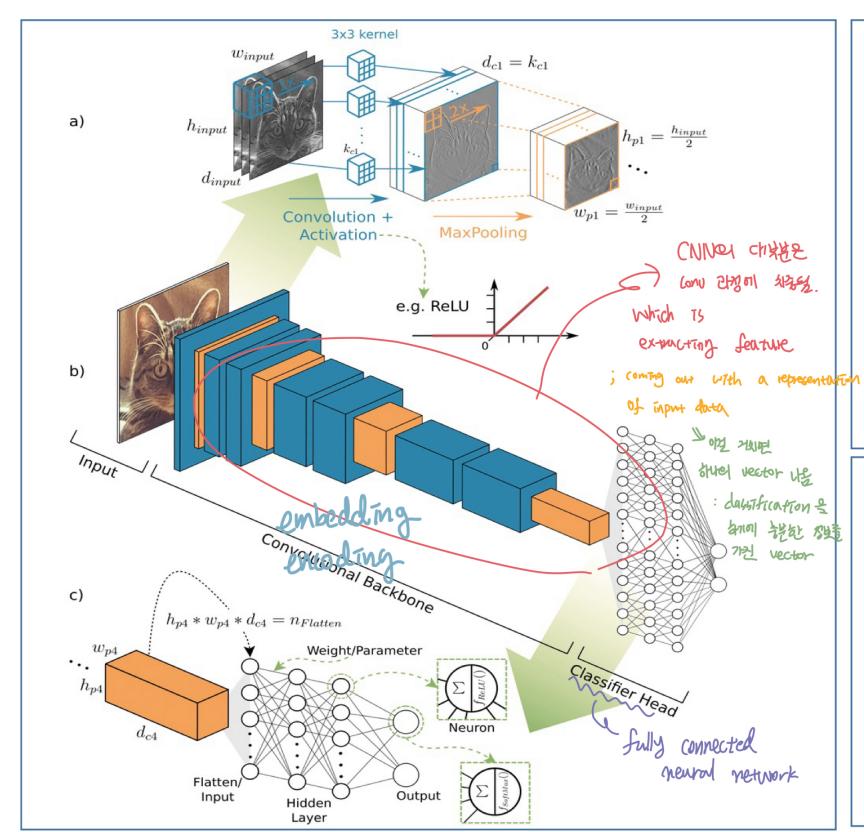


## FC layer

Fully Connected Network

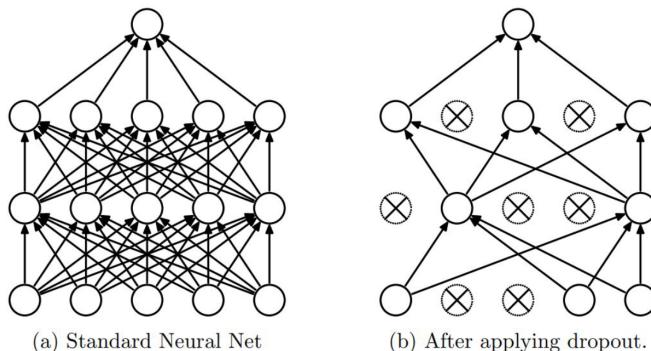
Linearly separable classifier

- cnn으로 feature extract 하고 FC layer로 classify



## dropout

오버피팅 방지를 위해 NN이 학습중일 때, 랜덤하게 뉴런을 꺼내 학습을 방해하는 기법.



[Lecture12.pdf](#)

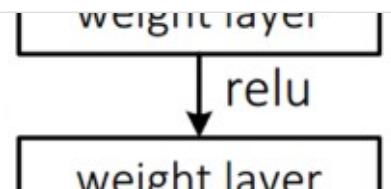
[Lecture13.pdf](#)

## ResNet

(논문리뷰) ResNet 설명 및 정리

이전의 연구들로 모델의 layer가 너무 깊어질수록 오히려 성능이 떨어지는 현상이 생김을 밝혀냈다. 이는 gradient vanishing/exploding 문제 때문에 학습이 잘 이루어지지 않기 때문이다. gradient vanishing이란 layer가 깊어질수록 미분을 점점 많이 하기 때문에 backpropagation을 해도 앞의 layer일수록 미분값이 작아져

👉 <https://ganghee-lee.tistory.com/41>



그 외

- ConvNet = CNN
- Convolutional Layer = CONV?
- Convolutional Layer가 정확히 무엇? 어떤 layer? 과정?
- ReLU = activation function
- 여러 번의 convolutional layer = activation maps = feature maps
- activation maps에 activation function 적용함