

# Lecture 5: Convolutional Neural Networks

## 목차

1. Convolutional Neural Networks
  - Fully Connected Layer
  - Convolution Layer
  - Pooling Layer

## Convolutional Neural Networks

이제 본격적으로 CNN 에 대해서 공부한다. CNN 은 이전의 많은 Neural Network 를 설계 하려던 작업을 이어받아 2012 년에 발표되게 되었다.

현재 CNN 은 많은 분야에서 사용되며 내가 관심있어하는 Image Understanding 의 영역 까지 사용된다.







| No errors   | Minor errors  | Somewhat related   | Image Captioning   |
|---|---|--|--|
|  |  |  | <p>[Vinyals et al., 2015]<br/>[Karpathy and Fei-Fei, 2015]</p>   |
| <i>A white teddy bear sitting in the grass</i>                                      | <i>A man in a baseball uniform throwing a ball</i>                                  | <i>A woman is holding a cat in her hand</i>  |  |
|  |  |  |  |
| <i>A man riding a wave on top of a surfboard</i>                                    | <i>A cat sitting on a suitcase on the floor</i>                                     | <i>A woman standing on a beach holding a surfboard</i>                               | <p>All images are CC0 Public domain:<br/><a href="https://pixabay.com/en/luggage-antique-cat-1643010/">https://pixabay.com/en/luggage-antique-cat-1643010/</a><br/><a href="https://pixabay.com/en/teddy-plush-bears-cute-teddy-bear-1623436/">https://pixabay.com/en/teddy-plush-bears-cute-teddy-bear-1623436/</a><br/><a href="https://pixabay.com/en/surf-wave-summer-sport-literal-1668716/">https://pixabay.com/en/surf-wave-summer-sport-literal-1668716/</a><br/><a href="https://pixabay.com/en/woman-female-model-portrait-adult-983967/">https://pixabay.com/en/woman-female-model-portrait-adult-983967/</a><br/><a href="https://pixabay.com/en/handsland-lake-meditation-496008/">https://pixabay.com/en/handsland-lake-meditation-496008/</a><br/><a href="https://pixabay.com/en/baseball-player-shortstop-infield-1045263/">https://pixabay.com/en/baseball-player-shortstop-infield-1045263/</a></p> <p>Captions generated by Justin Johnson using <a href="#">NeuralTalk2</a></p> |

Image Captioning

우선 CNN 의 구조에 대해 살펴본다.

---

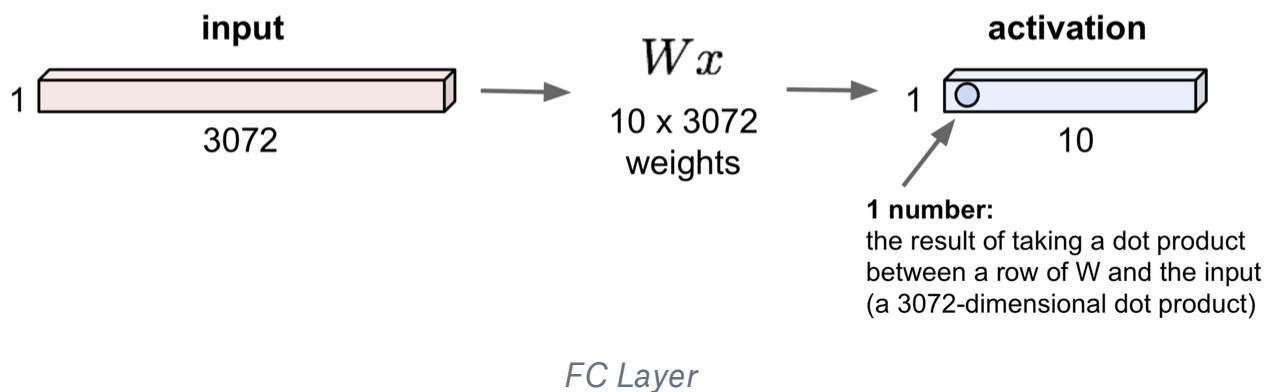
## Fully Connected Layer

FC Layer 은 우리가 이제까지 배워온 것을 나타낸다.

Input (1 x 3072) 와  $W(10 \times 3072)$  가 주어진다면 이를 통해 activation(score, 1 x 10) 으로 표현된다.

# Fully Connected Layer

32x32x3 image -> stretch to 3072 x 1

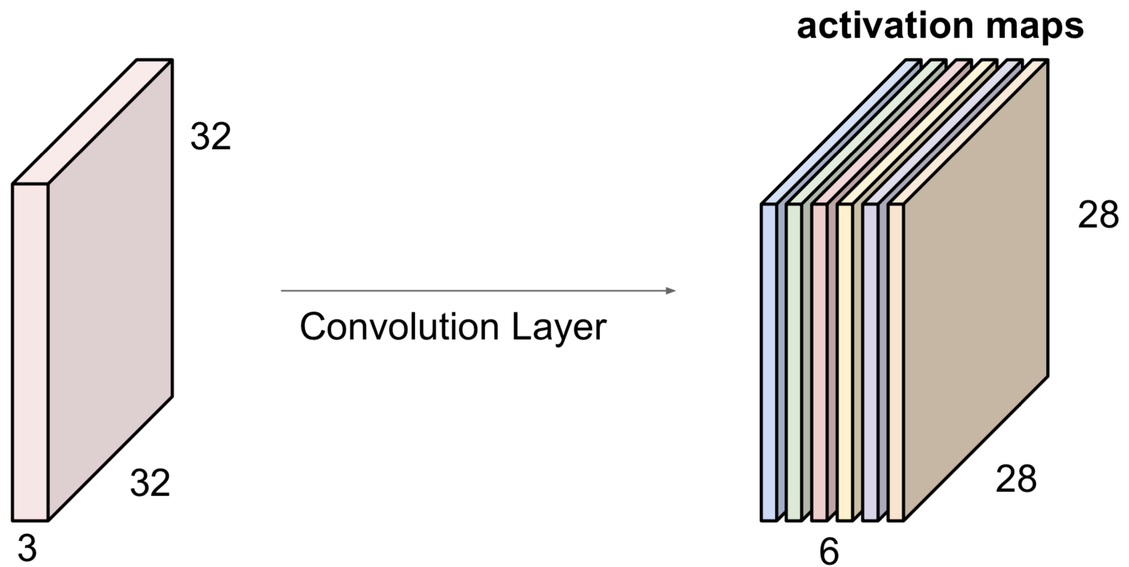


## Convolution Layer

Conv Layer (합성곱) 은 주어진 이미지에서 filter 에 맞게 feature map 을 뽑아내는 역할을 한다.

이때 **filter** 는 항상 input image 의 depth 를 따라간다.

아사실을 몰라서 강의듣는 동안 의문점어 참 많았다.



Ex 1

위의 그림처럼  $32 \times 32 \times 3$  의 이미지에  $5 \times 5$  의 filter 를 6개 적용하면  $28 \times 28 \times 6$  의 output 이 생성된다.

이때 이미지의 좌상단부터 filter 와의 내적을 통해 나온 결과가 output 각각 맵핑된다.

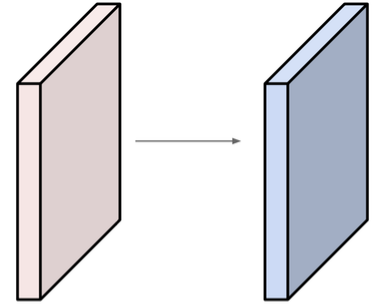
이때 우리는 stride 와 padding의 개념을 적용하게 되는데, stride 는 filter 를 움직일때 몇 칸씩 움직이는지를 나타내고 padding 은 각 꼭지줄의 데이터손실을 막기위해 0의 값을 각 꼭지줄 에 추가한다.

Padding을 예로들면  $32 \times 32 \times 3$  에 2의 패딩을 주게되면  $(32 + 2 * 2) \times (32 + 2 * 2) \times 3$  의 이미지를 이용하여 Conv를 진행한다.

Examples time:

Input volume: **32x32x3**

**10** **5x5** filters with stride 1, pad 2



Number of parameters in this layer?

each filter has  $5*5*3 + 1 = 76$  params (+1 for bias)

=>  $76*10 = 760$

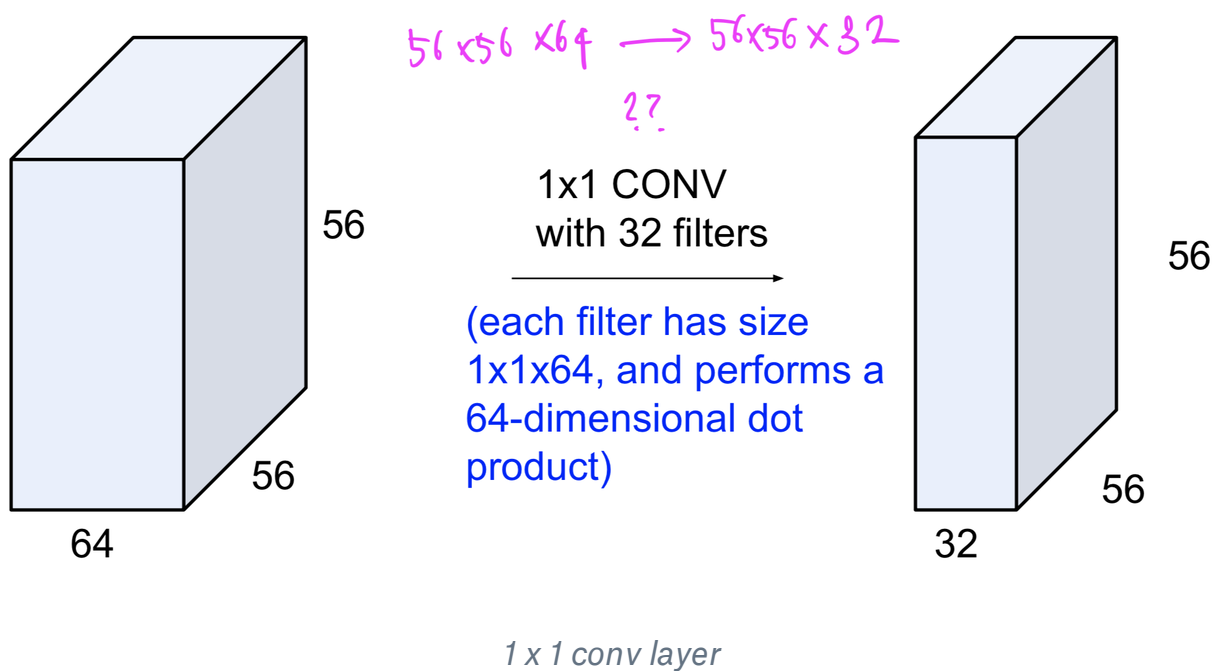
#### *Conv Layer*

우리는 위의 이미지처럼 32 x 32 x 3 의 이미지가 주어지고 10개의 5 x 5 filter 가 stride 1, pad 2 로 주어졌을 때, parameter 의 수를 filter의 크기 (25 x 25 x 3) + bias (1) 의 10개 filter 를 가진 것으로 파악할 수 있다.

이때 output 의 크기는 spatially  $(32 + 2 * 2 - 5) / 1 + 1$  로 32 x 32 x 10 을 가진다.

이번에는 1 x 1 Conv Layer를 적용하는 예를 보자.

(btw, 1x1 convolution layers make perfect sense)



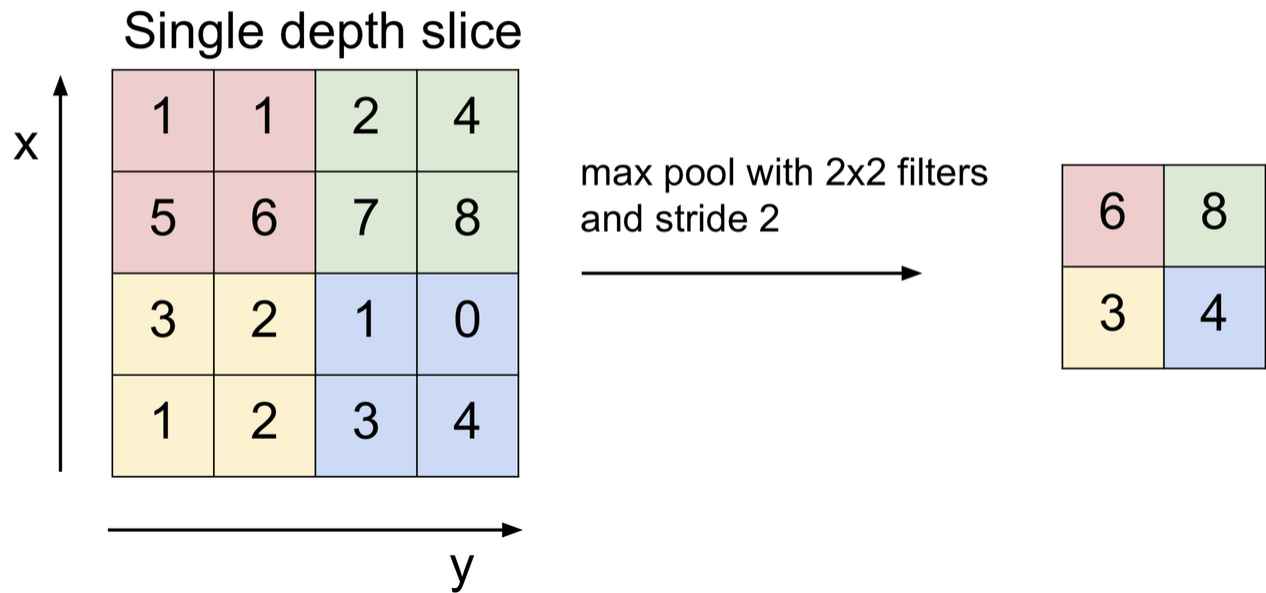
56 x 56 x 64 에 1 x 1 conv layer를 32개 적용하면 결과는 56 x 56 x 32 가 나온다.  
이는 이미지의 크기는 줄이지 않고 각 성분을 보존하면서 filter 를 통과하게 만드는 것이다.

---

## Pooling Layer

Pooling 은 일반적으로 이미지를 downsampling 하기 위해 사용된다.  
이때 이미지의 크기만 줄일 뿐 depth 는 건들이지 않는다.

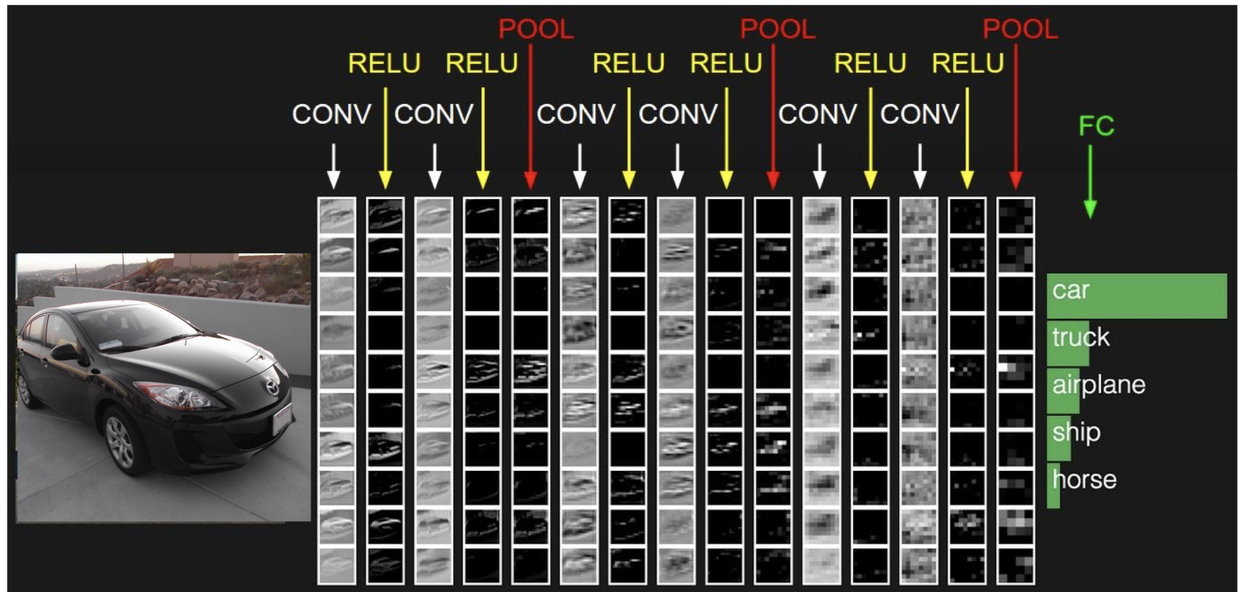
# MAX POOLING



## Max Pooling

이 강의에서 소개되는 방식은 max pooling 이다. 이는 pooling filter의 크기에서 max 값을 output 으로 넘겨주는 방식으로 이용된다. 이때 일반적으로 stride는 filter 의 크기만큼 주어져서 filter 를 이동할때 겹치는 부분이 없도록 한다.

강의에서 왜 max 값을 뽑아내는가에 대한 질문이 있다. 답변으로는 일반적으로 신호의 이미지가 주어졌을때 우리는 그 신호가 얼마나 활성화 되었는지를 보고싶다. 때문에 max pooling 을 사용하여 가장 강력한 신호를 뽑아낸다.



전체적인 CNN의 구조

일반적으로 CNN 은 위의 그림과 같이 CONV 로 이미지를 통과시키고 RELU 함수를 통해 activation map을 만들고 이를 반복적으로 하면서 pooling 을 통해 이미지의 크기를 줄여주며 마지막으로 Fully Connected Layer 을 통하여 Classification 을 적용한다. CNN 은 우리 뇌가 무엇인가를 인식하는 과정을 수학적으로, 코드로 구현한 것이고 이를 보완하는 여러 모델들이 앞으로 다루어질 예정이다.

#### Jun's Note

1. 무엇을 배웠는지  
| CNN 의 구조와 원리에 대해서 학습하였다.
2. 어디에 쓰이는지  
| 주어진 input 의 거대한 영역을 잘게 쪼개면서 인식하고 이를 계속 합친다. 마치 뉴런의 행동처럼.
3. 수식의 목적을 잘 파악했는지?  
| 특별히 수식이 들어가지는 않았으나 기본적인 CNN의 구조는 filter를 sliding 함에 있으므로 목적을 파악했다.