

## 2. 케이스 스터디

날짜 @2023년 12월 28일

### ▼ 목차

[왜 케이스 스터디를 하나요?](#)

[고전적인 네트워크들](#)

[1 LeNet-5](#)

[2 AlexNet](#)

[3 VGG-16](#)

[Residual Networks \(ResNets\)](#)

[1 Main Path in ResNet](#)

[2 Residual Block & Skip Connection](#)

[3 Train Error w.r.t. #layers](#)

[왜 ResNets가 잘 작동할까요?](#)

[Network 속의 Network](#)

[Inception 네트워크의 아이디어](#)

[1 The Problem of Computational Cost](#)

[2 Using 1×1 Convolution](#)

[Inception 네트워크](#)

[1 Inception Module](#)

[2 Inception Network](#)

[출석퀴즈 오답노트](#)

## 왜 케이스 스터디를 하나요?

합성곱 신경망을 구축하기 위해서는 효율적인 신경망 구조를 살펴봐야 합니다. 하나의 컴퓨터 비전 작업에서 잘 작동한 구조가 다른 작업에도 유용하고 잘 작동하기 때문입니다.

- 하나의 컴퓨터 비전 작업에서 잘 작동하는 신경망의 구조는 다른 작업에서도 잘 작동하는 경우가 많음
  - 하나의 신경망 구조가 고양이/개/사람 인식에 유용할 때 이를 자율주행차를 개발하는 등의 다른 컴퓨터 비전 작업에도 적용 가능
- 대표적인 신경망
  - Classic Networks
    - LeNet-5, AlexNet, VGGNet
  - ResNet

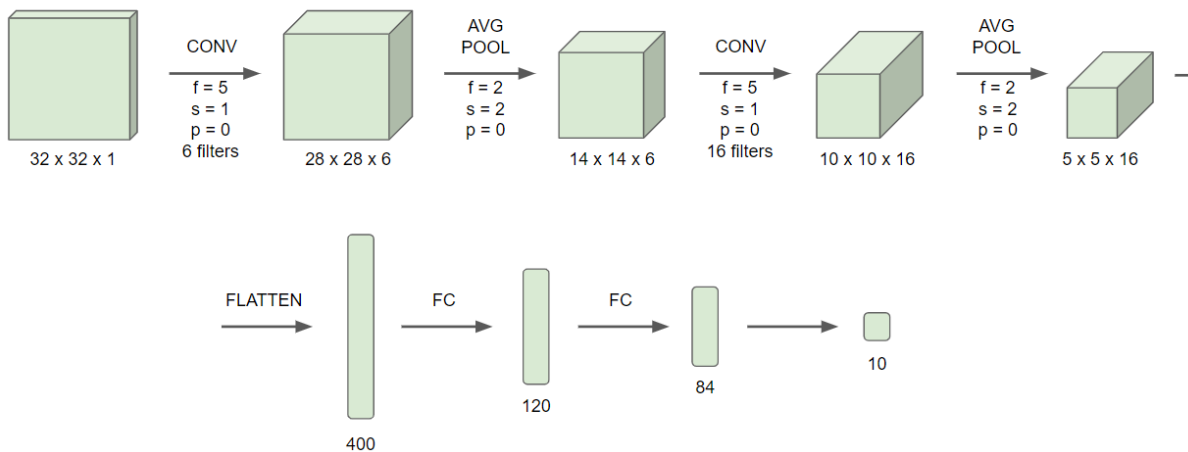
- 152개의 층을 훈련시킴
- Inception

## 고전적인 네트워크들

### 1 LeNet-5

- ◆ 흑백 이미지에 훈련되었으며, 손글씨의 숫자를 인식하는 것이 목적

#### LeNet - 5

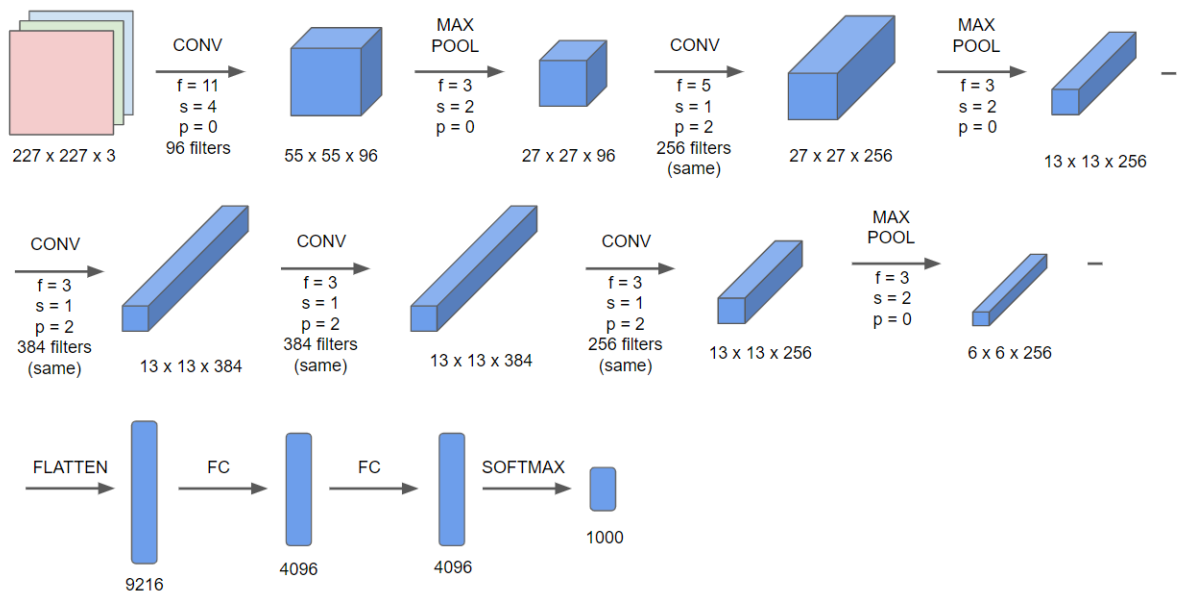


- 최대 풀링보다 평균 풀링이 더 보편적이었으며, 본 예시에서도 평균 풀링이 사용됨
- 패딩을 사용하지 않거나 유효 합성곱을 사용함
  - 합성곱 층을 적용할 때마다 높이와 너비가 감소함
- 변수의 개수가 상대적으로 적음 (60,000개)
- 풀링층 뒤에 비선형함수를 적용하였으며, 시그모이드 또는 tanh 함수를 사용함

### 2 AlexNet

- ◆ 이미지를 1,000개 클래스로 분류하는 것이 목적

## AlexNet



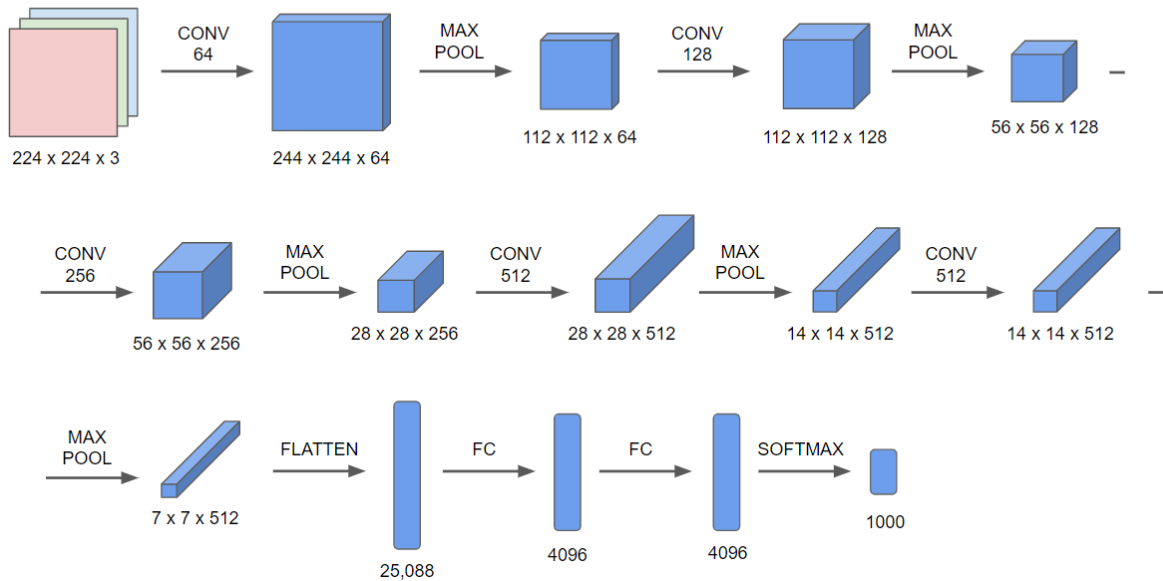
- LeNet-5과 비교했을 때 크기가 매우 큼
  - 변수의 개수가 약 6,000만 개
  - 더 많은 은닉 유닛과 데이터를 통해 훈련시켜 뛰어난 성능을 보임
- ReLU 활성화함수를 사용
- 지역 응답 정규화 층(Local Response Normalization)이 존재함
  - 예를 들어 크기가  $13 \times 13 \times 256$  일 때, LRN은 높이와 너비가 지정된 한 지점의 256개의 모든 채널을 보고 정규화함
  - 이후의 연구에서 유용하지 않음이 밝혀져 자주 사용되지 않음

## 3 VGG-16

## VGG-16

CONV : 3 x 3 filter, s=1, same

MAX POOL: 2 x 2, s=2

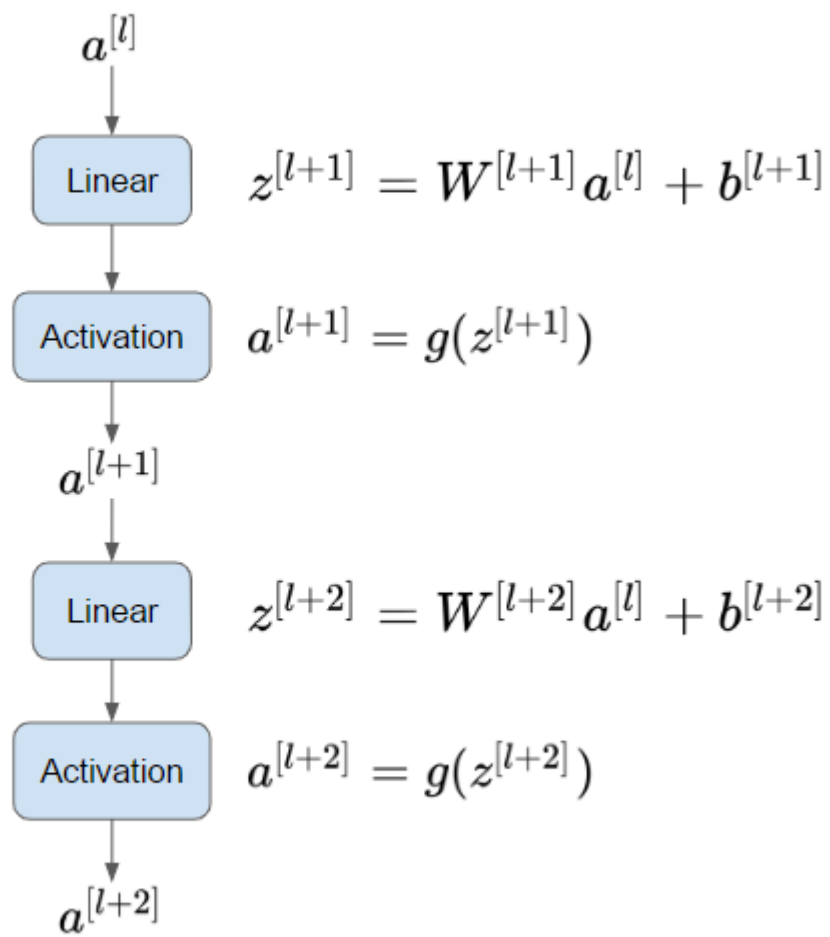


- AlexNet에 비해 간결한 구조를 가짐
  - 합성곱층의 필터의 개수가 64, 128, 256, 512로 2배씩 증가함
  - 훈련시킬 하이퍼파라미터의 수가 많아져 네트워크의 크기가 커진다는 단점이 있음
- 모든 합성곱 연산이 3x3의 필터를 가지며, 패딩 크기는 2, 스트라이드는 1임
- 2x2 픽셀씩 최대 풀링을 하며, 스트라이드는 2임

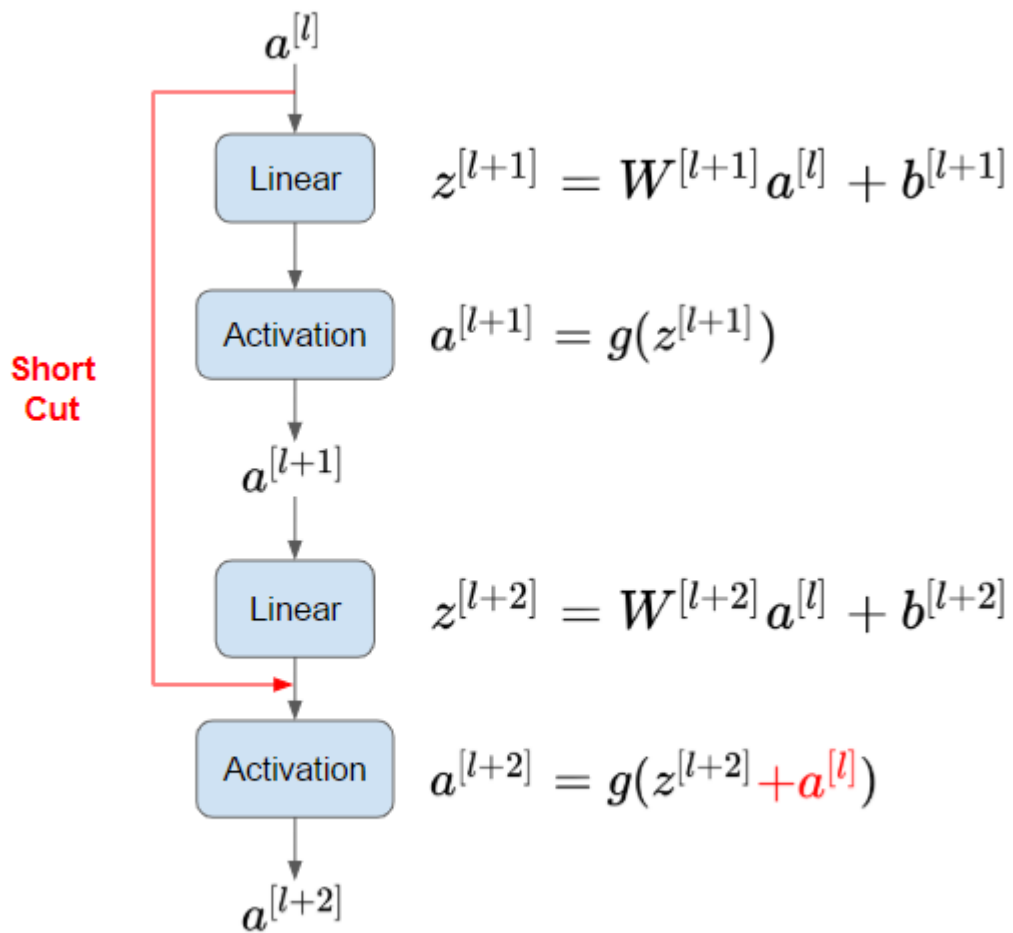
## Residual Networks (ResNets)

### 1 Main Path in ResNet

- 모든 층을 지나는 연산 과정을 main path라고 부름
  - $a^{[l]}$ 의 정보가  $a^{[l+2]}$ 로 흐르기 위해서는 모든 과정을 거쳐야 함

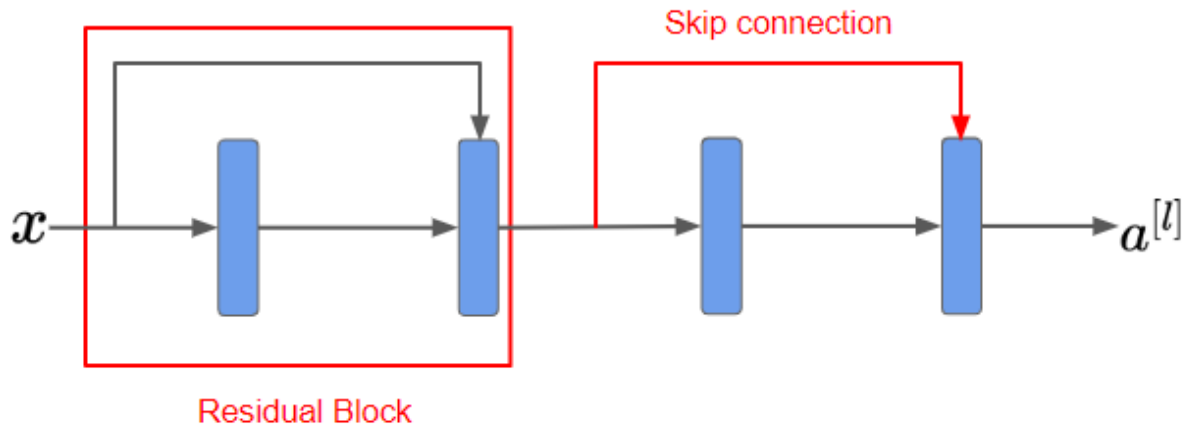


- ResNet에서는 다음과 같이  $z^{[l+2]}$ 에 비선형함수를 적용하기 전에  $a^{[l]}$ 를 더해줌



## 2 Residual Block & Skip Connection

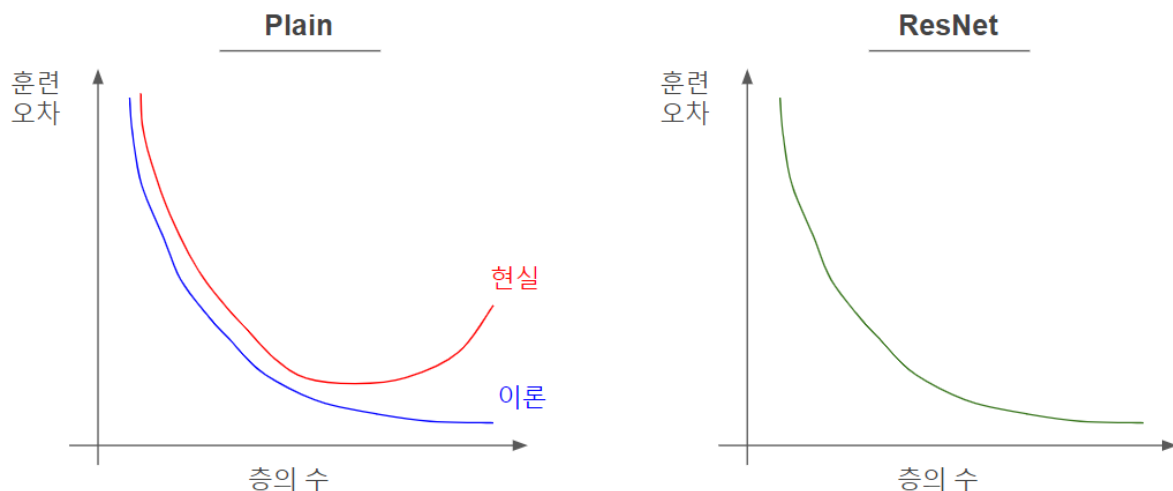
- ◆ ResNet은 여러 개의 잔여 블록으로 구성되어 있으며, 평행망에 스킵 연결을 더한 구조임



- 잔여 블록:  $a^{[l]}$ 를 더해서 다시 활성화 함수에 넣는 부분까지
- 스킵 연결:  $a^{[l]}$ 의 정보를 더 깊은 층으로 전달하기 위해 일부 층을 뛰어 넘는 역할입니다.

### 3 Train Error w.r.t. #layers

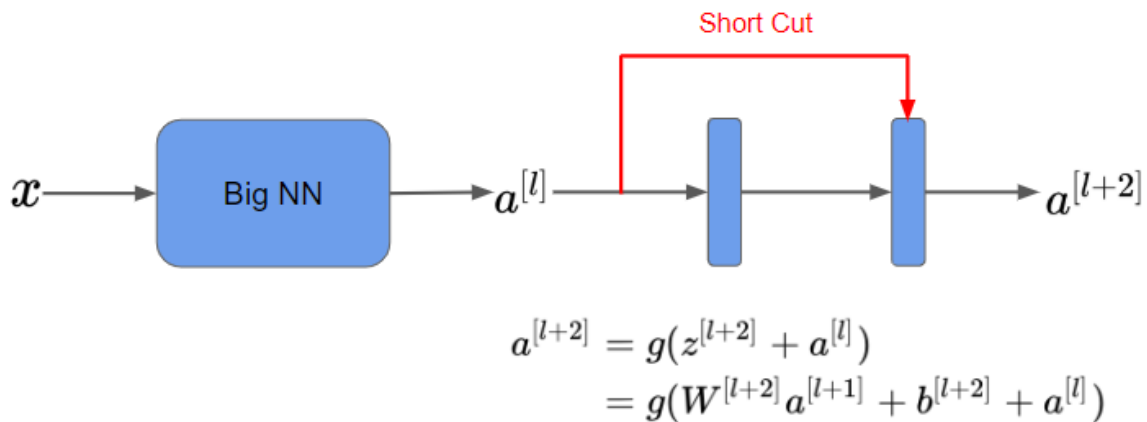
- ◆ 아주 깊은 신경망의 경우 경사가 소실되거나 폭발적으로 증가하는 문제가 발생하여 훈련시키기 어려운데, 스킵 연결을 활용하여 이를 해소할 수 있음



- 이론상으로는 신경망이 깊어질수록 훈련 오류는 계속 낮아져야 하지만, 경험적으로 층의 개수를 늘릴수록 훈련 오류는 감소하다가 다시 증가함
- 반면 ResNet에서는 스킵 연결을 활용하여 훈련 오류가 계속 감소하는 것을 확인할 수 있음

## 왜 ResNets가 잘 작동할까요?

- 다음과 같은 큰 신경망에 2개의 층을 추가한 뒤 스킵 연결을 더해줌
  - 이때 활성화함수  $g$ 는 ReLU를 사용함

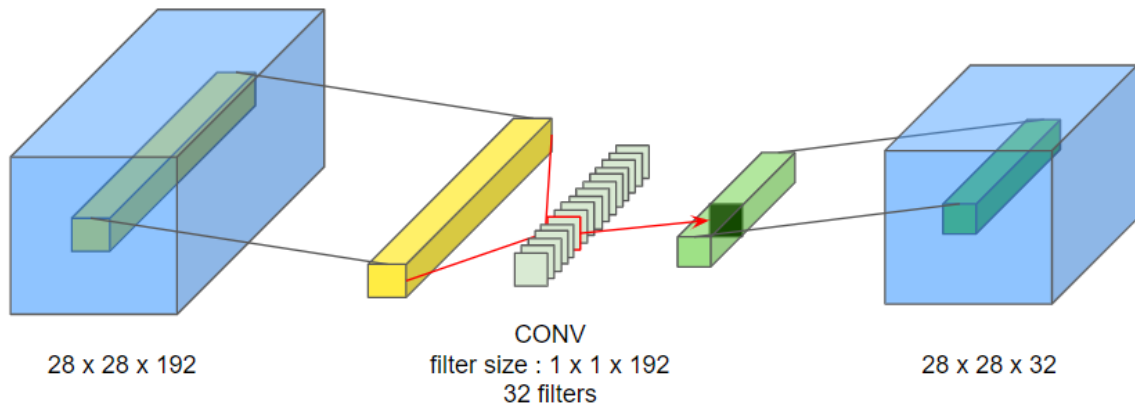


- 스킵 연결을 더해준 출력값  $a^{[l+2]}$ 은  $g(z^{[l+2]} + a^{[l]}) = g(W^{[l+2]} a^{[l+1]} + b^{[l+2]} + a^{[l]})$ 로 쓸 수 있음
- 이때  $W^{[l+2]}$ 와  $b^{[l+2]}$ 의 값이 0이 된다면, 위의 식은  $a^{[l+2]} = g(a^{[l]}) = a^{[l]}$ 으로 항등식이 됨
  - 위 항등식의 의미는 신경망으로 하여금 스킵 연결을 통해 두 층이 없는 더 간단한 항등식을 학습하여, 두 층 없이도 더 좋은 성능을 낼 수 있게 만들어주는 것임
  - 다만 이를 위해서는  $z^{[l+2]}$ 와  $a^{[l]}$ 이 같은 차원을 가져야 함
    - 일반적으로 동일합성곱 연산을 하거나 차원을 같게 만들어주는 행렬  $W_s$ 를 잔여 블록 앞에 곱해줌

## Network 속의 Network

- ◆ 합성곱 신경망을 구축할 때  $1 \times 1$  합성곱은 매우 유용함



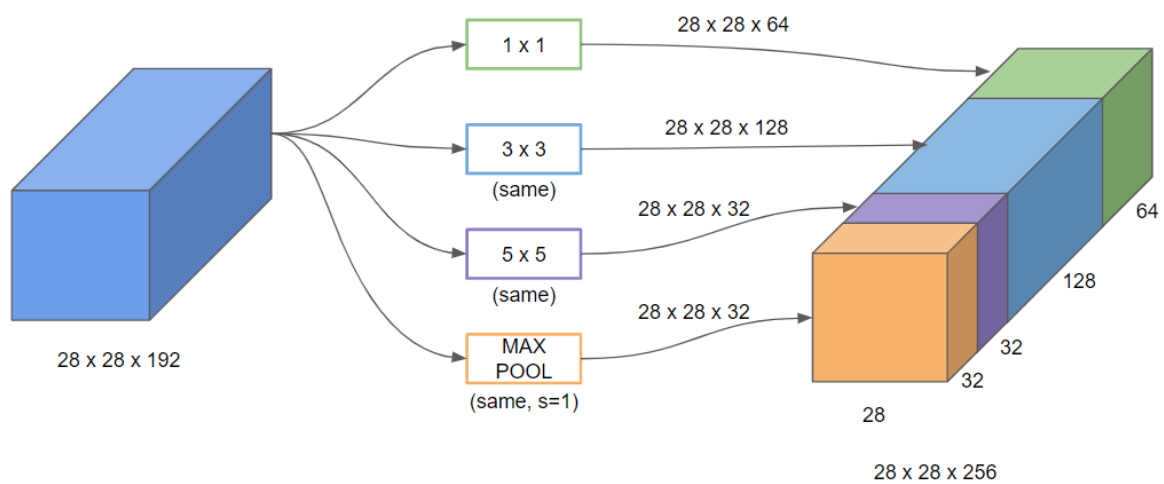


- 192개의 입력숫자가 32개의  $1 \times 1$  필터와 합성곱을 하면 32개의 출력 숫자가 됨
  - 입력 채널의 수만큼 유닛을 입력으로 받아 이들을 하나로 묶는 연산과정을 통해 출력채널의 수만큼 출력하는 작은 신경망 네트워크로 간주할 수 있음 (네트워크 안의 네트워크)
- 이처럼  $1 \times 1$  합성곱 연산을 통해 비선형함수를 하나 추가해 더 복잡한 함수를 학습시키거나 채널 수를 조절해줄 수 있음

## Inception 네트워크의 아이디어

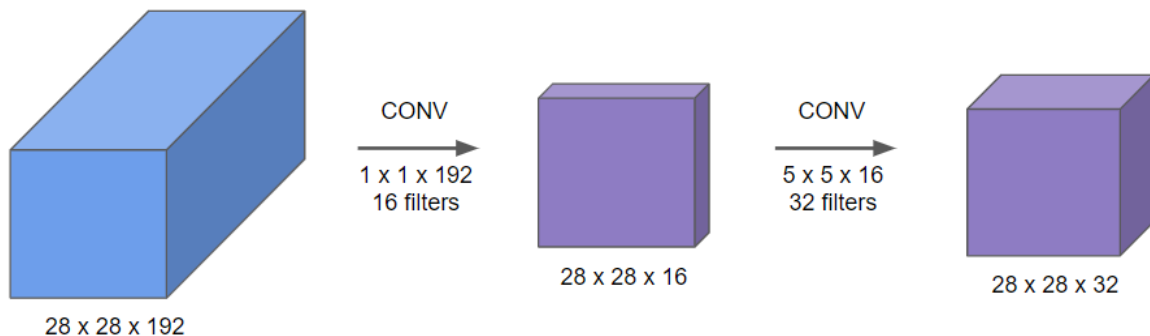
- ◆ 필터의 크기나 풀링을 따로 정하는 대신 모두 적용하여 그 출력들을 모두 합친 뒤, 네트워크로 하여금 변수나 필터 크기의 조합을 스스로 학습하도록 만드는 것이 핵심 아이디어임

### 1 The Problem of Computational Cost



- 계산 비용이 크다는 문제가 있음
  - 단순히  $5 \times 5$  필터를 사용한다고 가정해도  $28 \times 28 \times 32 \times 5 \times 5 \times 192 =$  약 1억 2,000만 개의 숫자를 계산해야 함
  - $1 \times 1$  합성곱으로 해결할 수 있음

## 2 Using $1 \times 1$ Convolution



- $5 \times 5$  합성곱을 사용하기 전  $1 \times 1$ 의 합성곱 연산을 통해 입력 이미지의 볼륨을 줄이는 작업을 함
- 그 후 다시  $5 \times 5$  합성곱 연산을 하며, 이때 계산 비용은 약 1,240만 개로 다음과 같음
  - $1 \times 1$  합성곱:  $28 \times 28 \times 16 \times 1 \times 1 \times 192 =$  약 240만 개
  - $5 \times 5$  합성곱:  $28 \times 28 \times 32 \times 5 \times 5 \times 16 =$  약 1,000만 개
- 학습에 필요한 계산 비용이 1/10 수준으로 크게 줄어든 것을 알 수 있음

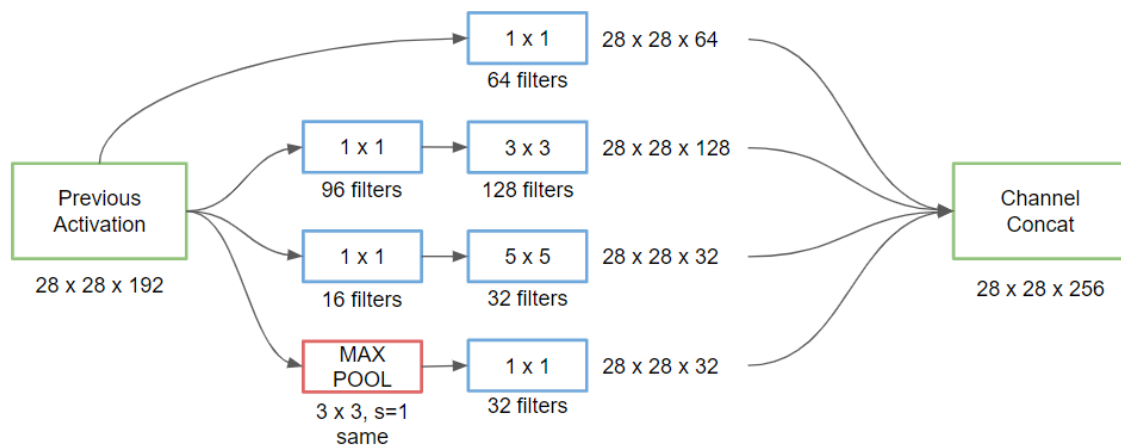
- 이때 사용된  $1 \times 1$  합성곱 층을 병목 층이라고 부름
- 병목 층 사용 시 표현의 크기가 줄어드는데, 적절하게 구현했을 경우에는 성능에 큰 지장 없이 많은 수의 계산을 줄일 수 있음

## Inception 네트워크

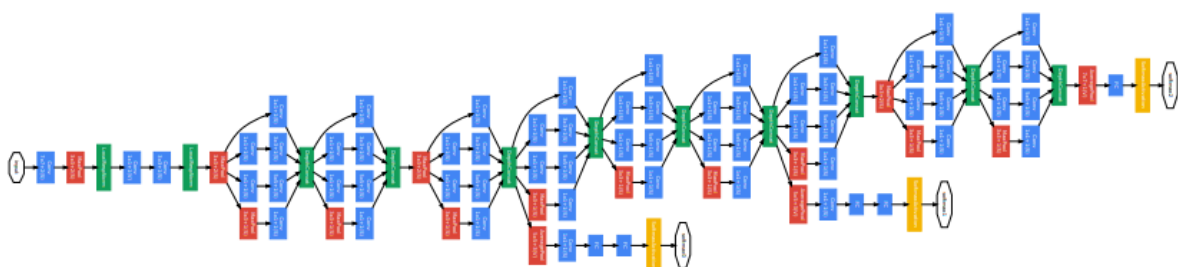
### 1 Inception Module

- 인셉션 네트워크는 다음과 같은 인셉션 모듈 여러 개로 구성되어 있음

### Inception Module



### 2 Inception Network



- 중간중간에 차원을 바꾸기 위해 최대 풀링 층과 여러 개의 인셉션 블록이 계속 반복되는 것을 볼 수 있음

## 출석퀴즈 오답노트

### ▼ 2. 고전적인 신경망 구조

ResNet(X), Inception(X), **VGG, AlexNet**

### ▼ 5. Residual Networks에서 스킵 연결이 도입되어 해결되는 문제

- 과적합 문제 (X)
- 경사 소실 또는 폭발 문제
- 아주 깊은 신경망을 학습하지 못하는 문제

### ▼ 6. ResNet에 대해 옳지 않은 설명

- 지름길은  $a^{[l]}$ 의 정보를 출력층으로 바로 전달하기 위한 역할이다
- ResNet은 main path는 사용하지 않는다
- ResNet은 깊은 신경망에 대해 좋은 성능을 발휘하지만 100층 이상의 신경망에서는 훈련오류가 증가한다