



# Deep Learning based Digital Image Processing

Hahyun Kim, Woobin Choi

Department of Applied Mathematics, Kyung Hee University

## Introduction

- 2016년 3월 구글과 딥마인드 가 만든 인공지능인 알파고와 이세돌 과 바둑으로 격돌하면서 세상에 등장.

- 알파고의 알고리즘 : 몬테카를로 탐색, CNN(Convolutional Neural Network).

-- Deep Learning의 알고리즘 중에 하나인 CNN은 대표적인 인공지능 알고리즘으로, 이미지 분류 문제에 주로 사용.

- 이미지 분류 : 입력 이미지를 미리 정해진 카테고리 중 하나인 라벨로 분류하는 것.  
➡ 컴퓨터 비전, 물체 검출, 영상 분할 등의 이미지 분류 문제를 해결.

## Models & Results

### 1. CNN Model

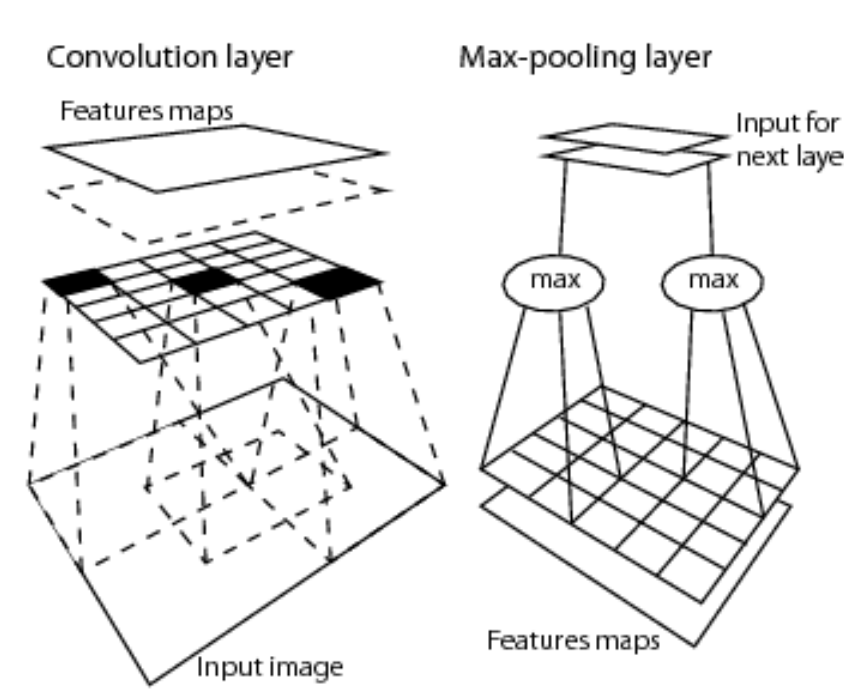
Convolution: signal processing 분야에서 아주 많이 사용하는 operation.

$$s(t) = (x * w)(t) = \int x(a)w(t - a)da.$$

: 주어진 데이터 x에 filter w 를 사용하여 데이터를 처리.



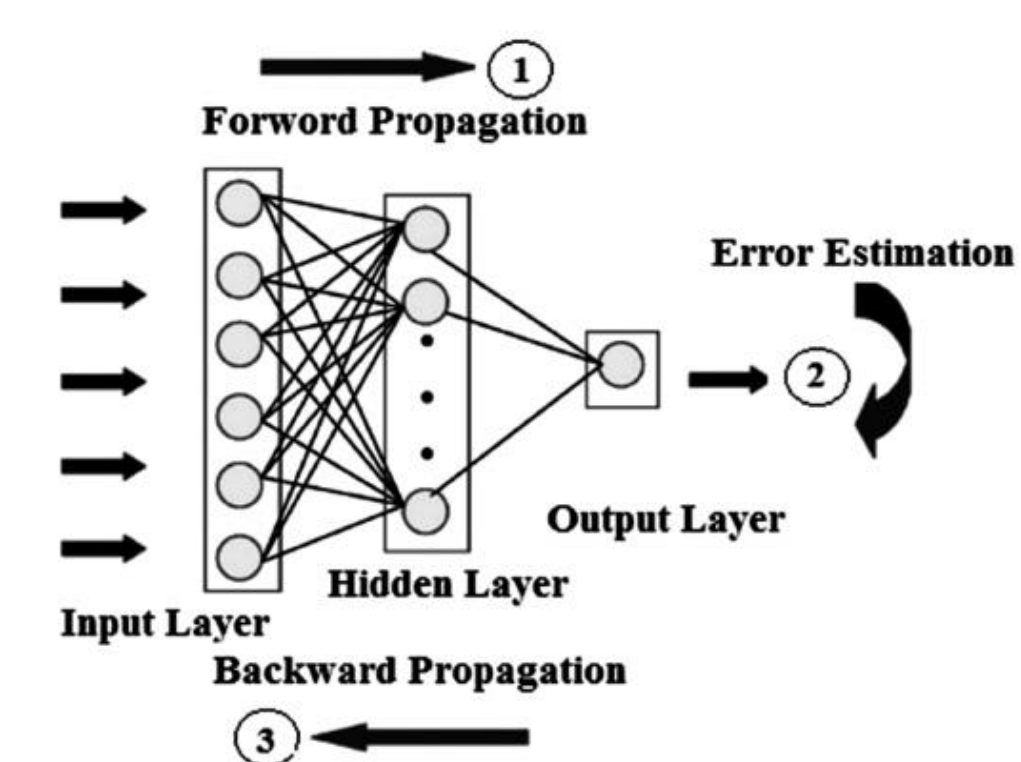
➡ 어떤 filter를 이용하여 주어진 image의 적절한 feature를 추출하는 알고리즘.



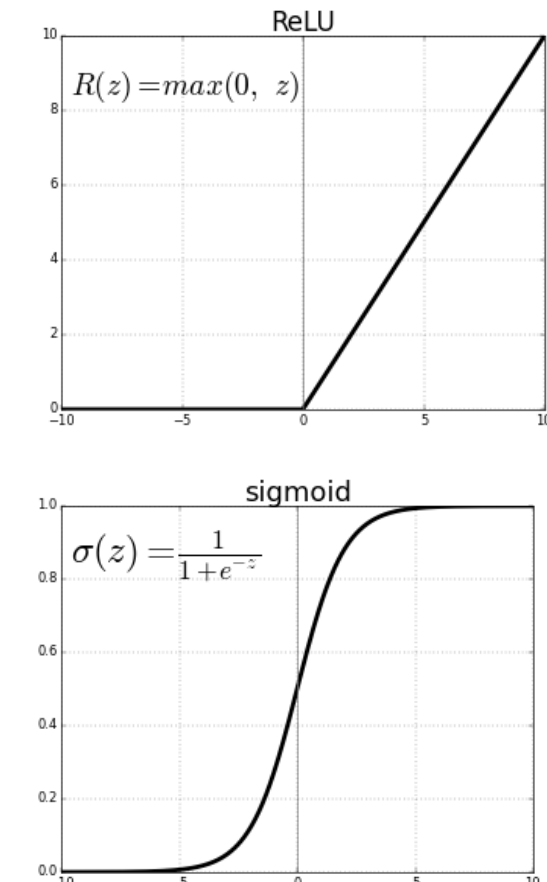
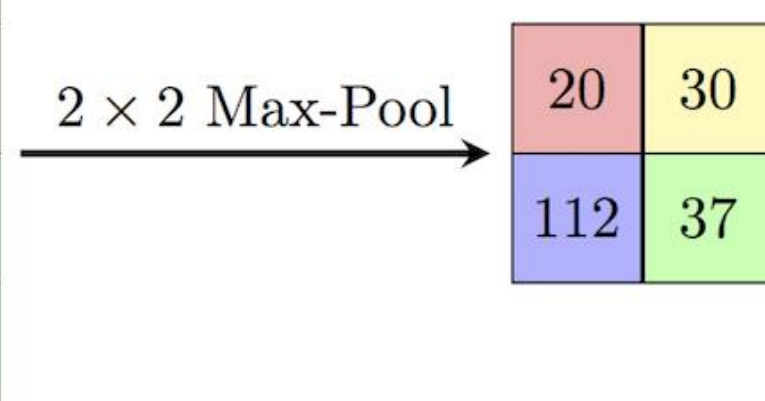
- Back propagation

- Max pooling layer

- Activation function



12	20	30	0
8	12	2	0
34	70	37	4
112	100	25	12

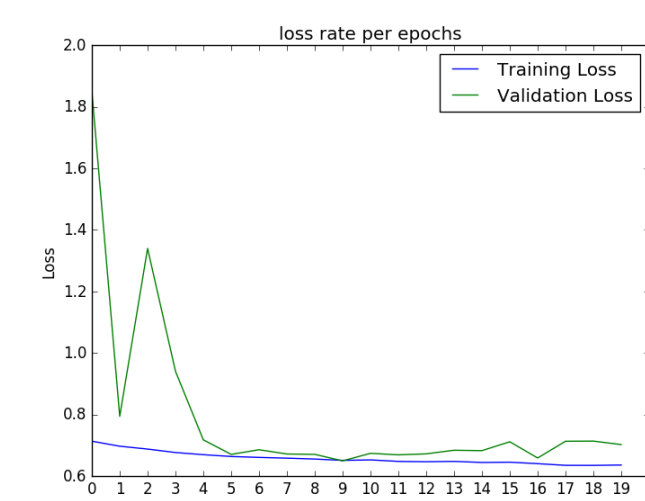


- data 속성 : 128 x 128 pixel, RGB.  
(1) 개와 고양이 각각 3,500장씩 훈련(총 7,000장).  
(2) Validation set으로 개와 고양이 각각 1,000장씩, 총 2,000장으로 정확도 검증.  
Accuracy : 60.08% /// 연산시간 : 16분 28초 (CPU).



<개선방향>

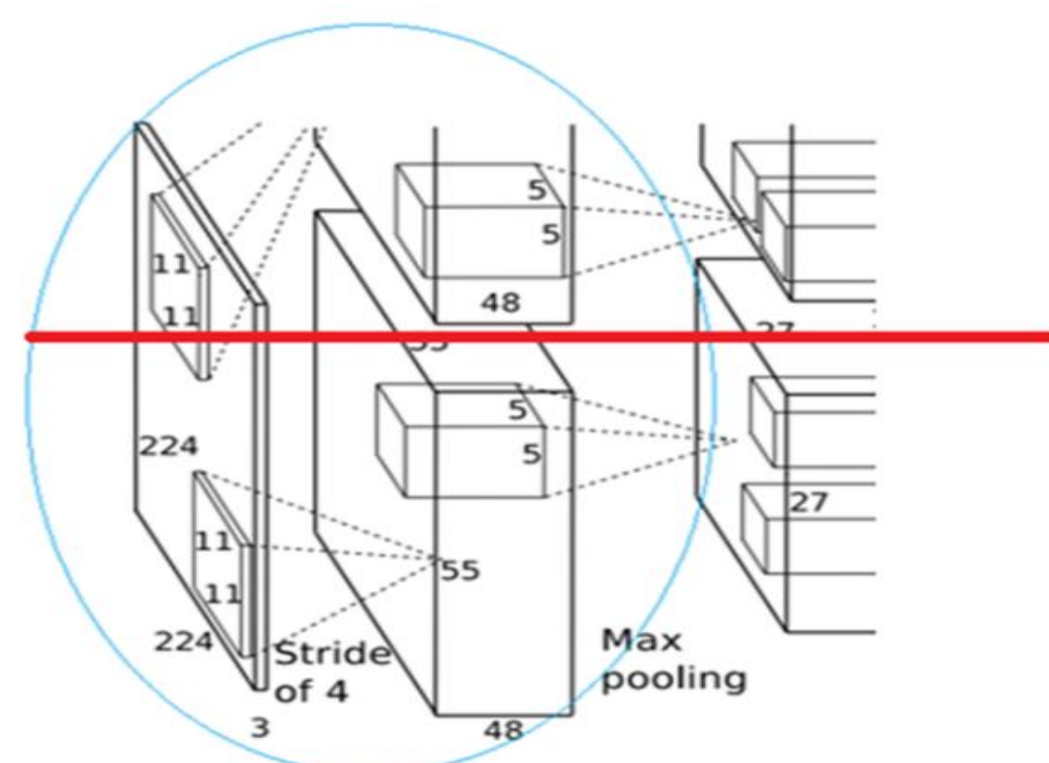
- data set 증가 : 총 7,000장 -> 15,000장.  
- Batch Normalization: 정규화 과정 중 하나, layer가 깊어 질때마다, 불안정한 layers들을, 평균 0, 분산 1로 정규화.  
Accuracy : 70.67% /// 연산시간 : 28분 4초 (GPU: GTX 1050 3GB).  
➡ tensorflow - gpu 이용.



### 3. Alex - net

ImageNet 영상 데이터 베이스를 기반으로 한, ILSVRC(ImageNet Large Scale Visual Recognition Challenge) - 2012 우승 한 CNN 알고리즘.

- 특징  
2개의 GPU를 기반으로 한 병렬 구조.



\* GPU - 1 : RGB와 상관 없는 정보를 추출하기 위한 kernel 학습.  
\* GPU - 2 : RGB와 상관 있는 정보를 추출하기 위한 kernel 학습.

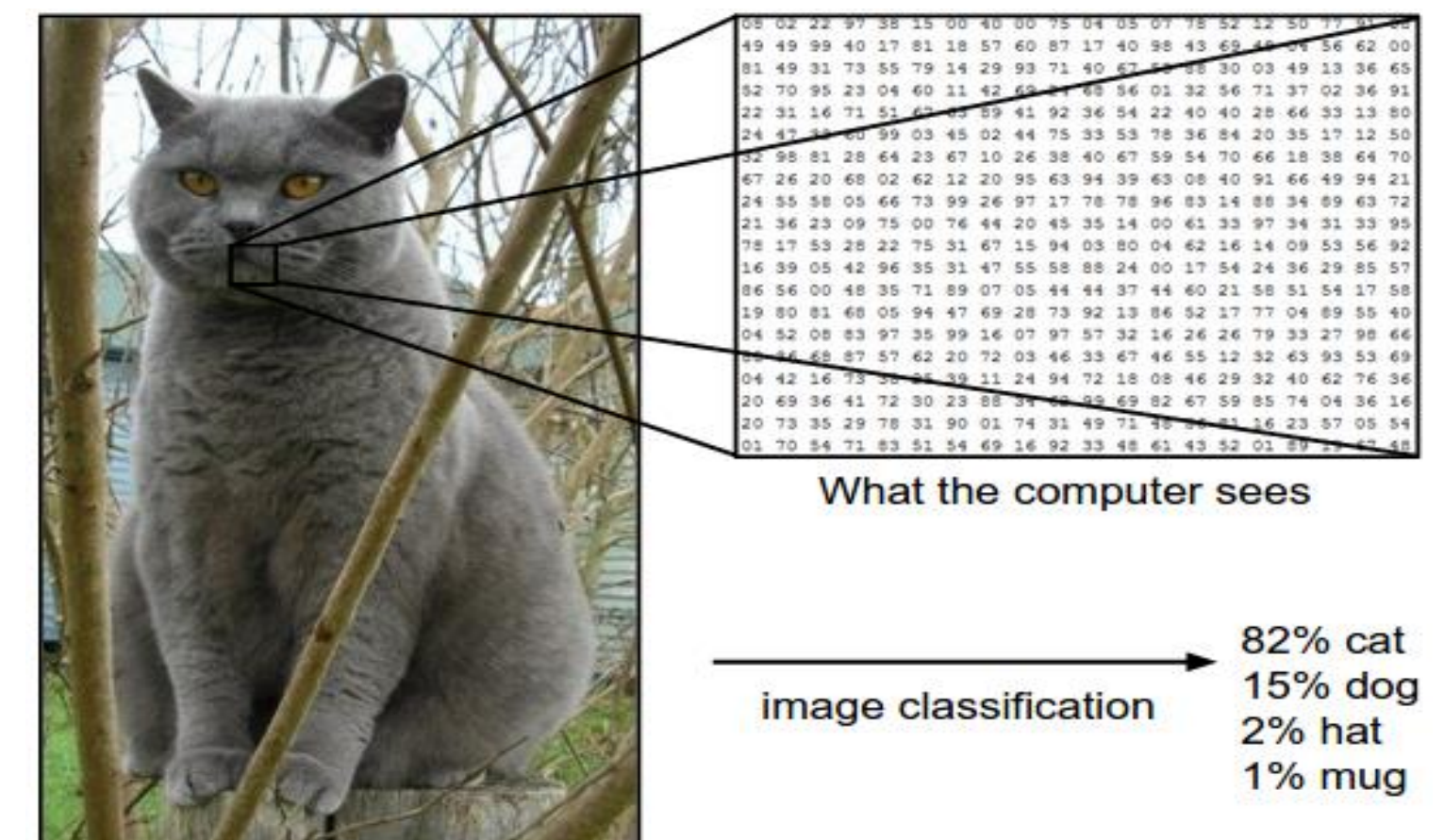
- Accuracy 측정 : validation set(개, 고양이 각각 976장씩 총 1954장).  
➡ 90.85%.

## Research & Abstract

- 이미지 분류의 기초적인 문제인 개와 고양이의 binary classification을 구현.

- Python 으로 CNN 에 사용되는 파라미터 들을 조정하여 정확도 검증을 위한 모델 구현.  
- Matlab 으로 검증된 모델인 Alex net , Google net을 같은 데이터에 적용하여, 정확도 검증.

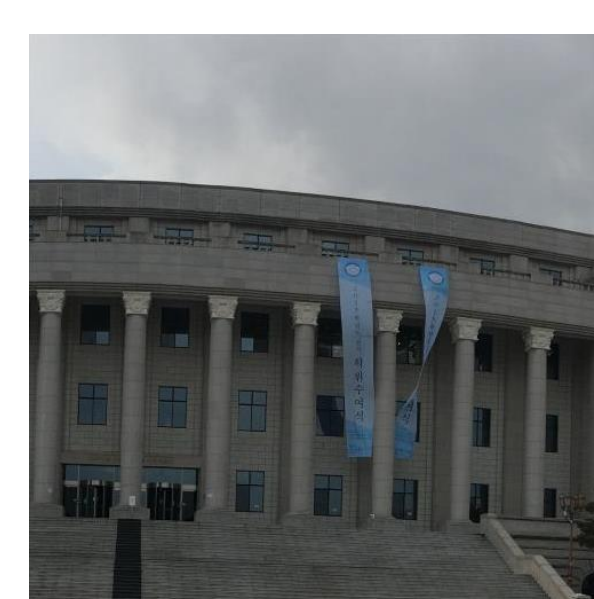
- 최종적으로 CNN의 실생활 적용 가능성 사실 입증.



82% cat  
15% dog  
2% hat  
1% mug

### 2. Wavelet Transform

임의의 함수(신호)를 시간-주파수 영역에서 동시에 국부성 을 만족하는 기저함수의 선형결합으로 표현하는 것. 이를 통해, 영상 압축이나 음성인식 등에 사용. 기저함수로 Haar Wavelet 사용.

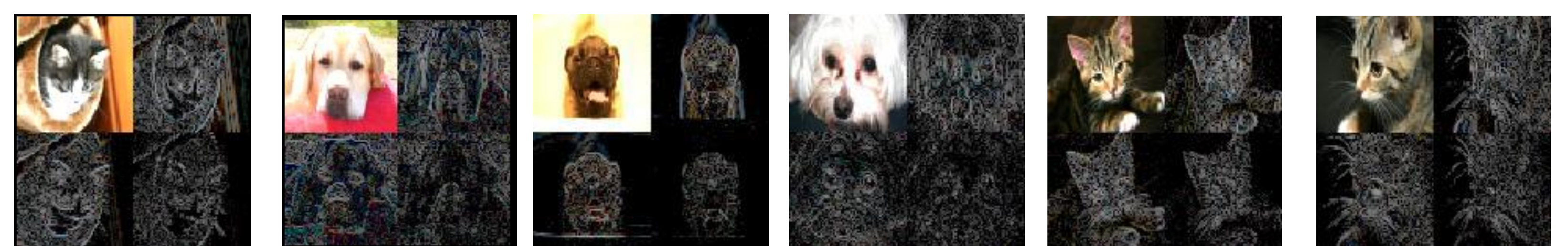


$$X = \begin{bmatrix} 1 & 97 & 87 & 86 & 104 & 109 & 111 & 109 & 109 & 113 & 33 & 18 \\ 1 & 56 & 97 & 100 & 105 & 100 & 91 & 90 & 90 & 103 & 21 & 11 \\ 7 & 107 & 105 & 113 & 109 & 94 & 95 & 111 & 105 & 44 & 9 & 14 \\ 3 & 99 & 102 & 96 & 99 & 97 & 108 & 119 & 124 & 39 & 10 & 13 \\ 2 & 102 & 98 & 102 & 107 & 113 & 116 & 120 & 123 & 34 & 8 & 14 \\ 2 & 52 & 80 & 74 & 84 & 95 & 104 & 113 & 121 & 101 & 41 & 10 \\ 3 & 81 & 79 & 68 & 47 & 38 & 41 & 44 & 50 & 59 & 39 & 11 \\ 2 & 89 & 87 & 89 & 79 & 65 & 40 & 41 & 35 & 10 & 9 & 17 \\ 3 & 89 & 88 & 91 & 87 & 72 & 39 & 16 & 17 & 19 & 23 & 40 \\ 2 & 87 & 87 & 88 & 84 & 74 & 41 & 35 & 40 & 43 & 36 & 18 \\ 1 & 65 & 87 & 86 & 86 & 80 & 49 & 27 & 39 & 40 & 8 & 11 \\ 3 & 87 & 85 & 86 & 88 & 82 & 95 & 27 & 33 & 40 & 10 & 11 \\ 1 & 86 & 87 & 80 & 88 & 81 & 50 & 8 & 10 & 10 & 14 & 13 \\ 9 & 65 & 88 & 89 &td="9"> 89 | 89 | 83 | 89 | 11 | 12 | 13 | 12 | 13 |  |  || 3 | 87 | 88 | 90 | 89 | 83 | 52 | 10 | 14 | 12 | 11 | 11 |
| 1 | 85 | 88 | 90 | 88 | 88 | 52 | 12 | 15 | 15 | 14 | 13 |
| 3 | 82 | 89 | 87 | 87 | 82 | 52 | 12 | 14 | 15 | 14 | 15 |
| 1 | 86 | 87 | 86 | 81 | 81 | 50 | 9 | 15 | 15 | 15 | 17 |$$

$$X' = \begin{bmatrix} H \\ G \end{bmatrix} X \begin{bmatrix} H \\ G \end{bmatrix}^T = \begin{bmatrix} HXH^T & HXG^T \\ GXH^T & GXG^T \end{bmatrix} = \begin{bmatrix} A & D_h \\ D_v & D_d \end{bmatrix}$$

좌 상단 : 수평방향과 수직방향 모두로부터 평균된 계수들을 포함.  
우 상단 : 영상의 수직방향 경계에 대한 정보로서 해석.  
좌 하단 : 영상의 수평방향 경계에 관한 정보.  
우 하단 : 대각선 방향 경계 표시.

Wavelet transform을 CNN에 적용하여 feature reduction을 통해 learning time을 줄일 수 있다. 그러나 정확도는 오히려 하락하였다.



### 4. Google - net

ILSVRC 2014 우승한 CNN 알고리즘 (1,200,000 images trained).

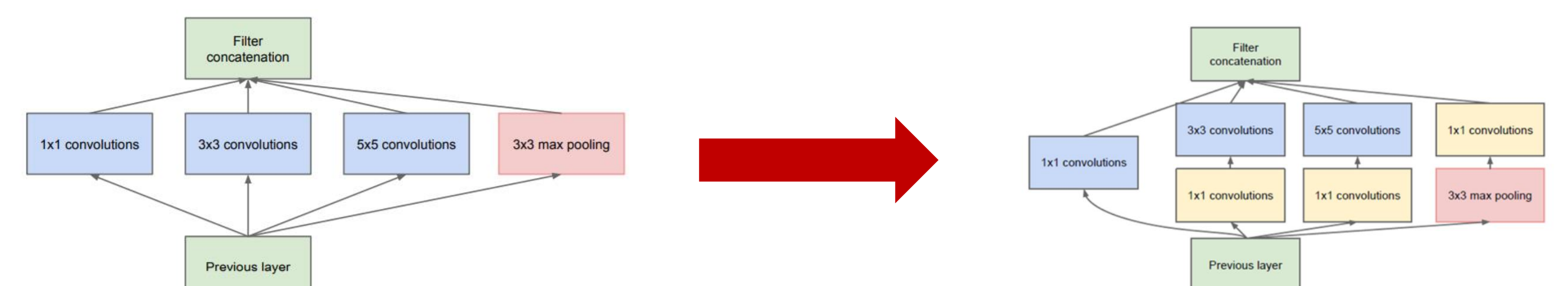
- 특징

Networks are deeper and deeper !!! ➡ I recommend GPU better than CPU because of time spending.

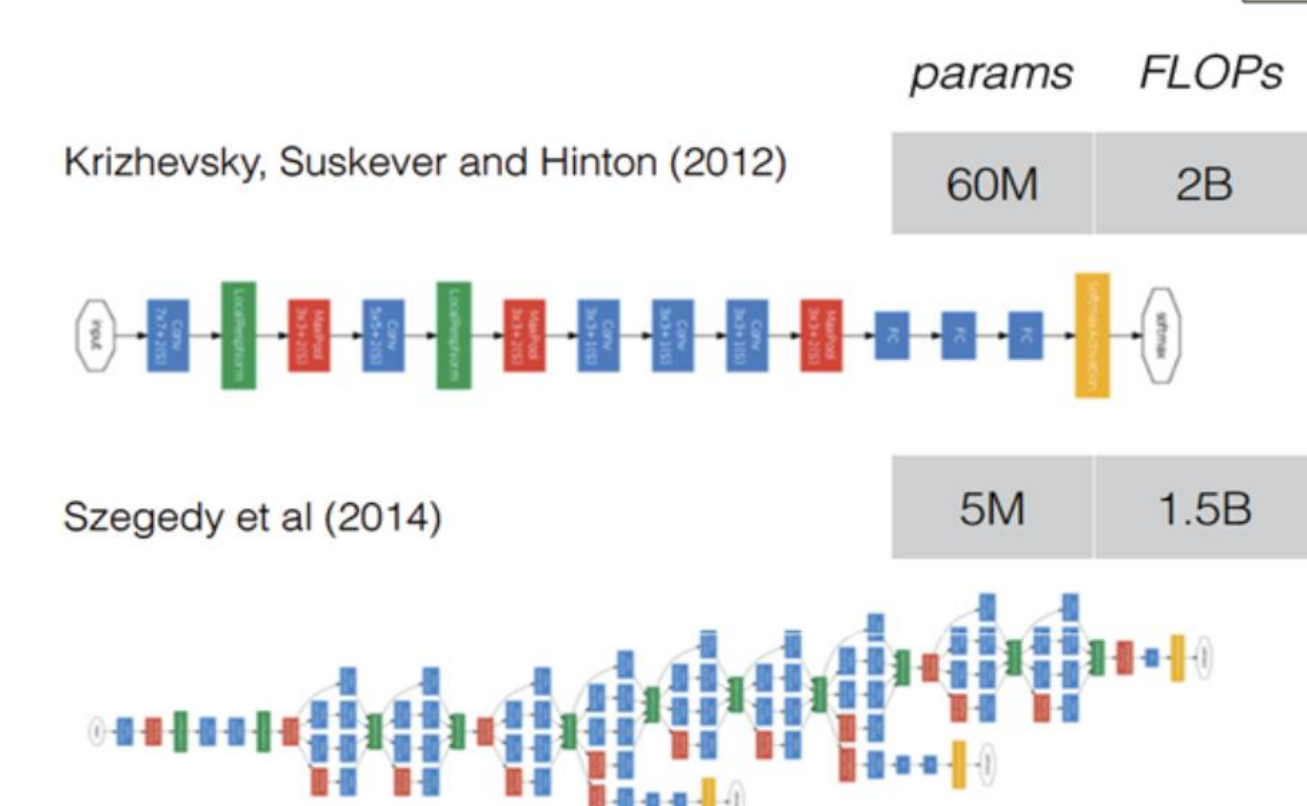
2013년도 까지는 10 layers 미만이었지만, Google net은 22 layers.

\*layer 증가 시 parameter 수 증가, 연산 량 증가.

➡ overfitting(과 적합) 발생 ➡ 해결책(Inception model)



- Alex - net vs Google - net



Accuracy 측정 : validation set(개와 고양이 각각 77장 씩, 총 154장).

➡ 93.33% (mini batch size : 32, validation split : 0.2).  
93.48% (mini batch size : 10 , validation split : 0.3).  
94.74% (mini batch size : 32, validation split : 0.25).

## Conclusions

- CNN 의 기초적인 문제인 binary classification 을 Python 으로 검증.  
- Wavelet Transform을 통한 learning time 단축, 그러나 정확도는 오히려 하락하였다.  
- Matlab 을 이용한 pretrained Alex - net , Google - net을 통한 정확도 향상.  
- 인공지능의 컴퓨터 비전, 영상인식 등 다양한 실생활 문제에 CNN 기술 접목 가능.  
- Capstone Design total process : [https://github.com/choiwb/Applied\\_Mathematics\\_Capstone\\_Design](https://github.com/choiwb/Applied_Mathematics_Capstone_Design) (related pdf, Matlab , Python code). If you have more question, please contact [choiwb1216@khu.ac.kr](mailto:choiwb1216@khu.ac.kr).  
<References>  
- Convolutional Neural Network Feature Reduction using Wavelet Transform, ELEKTRONIKA IR ELEKTROTECHNIKA, ISSN 1392-1215, VOL. 19, NO. 3, 2013  
- Going Deeper with Convolutions , IEEE Xplore  
- ImageNet Classification with Deep Convolutional Neural Networks , Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton