

Convolutional Neural Network Residual Network

POSTECH A.I
Seungbeom Lee

Table of Contents

- CNN
 - Review the key points
 - Practice
- ResNet
 - Review the key points
 - Practice

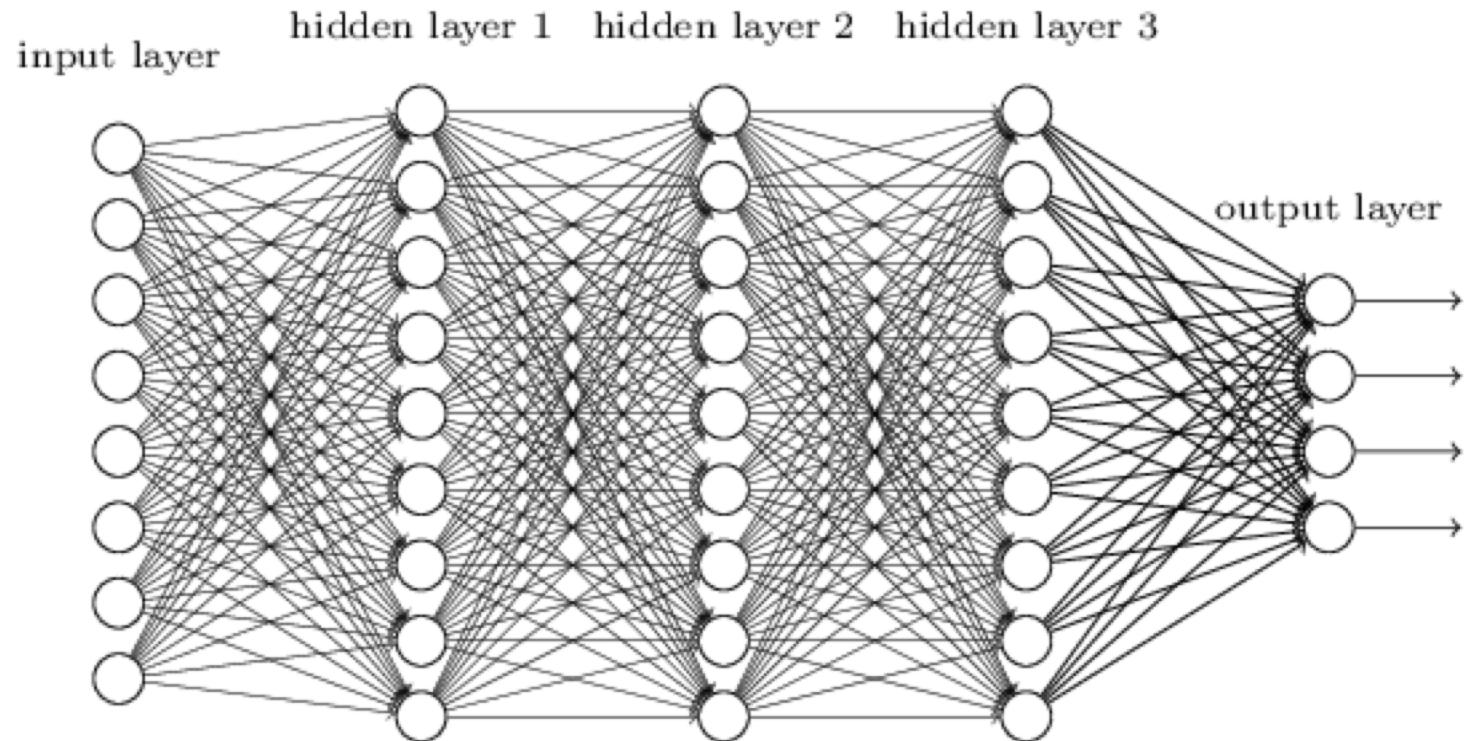
Review the key points

- Motivation of CNN – parameter sharing

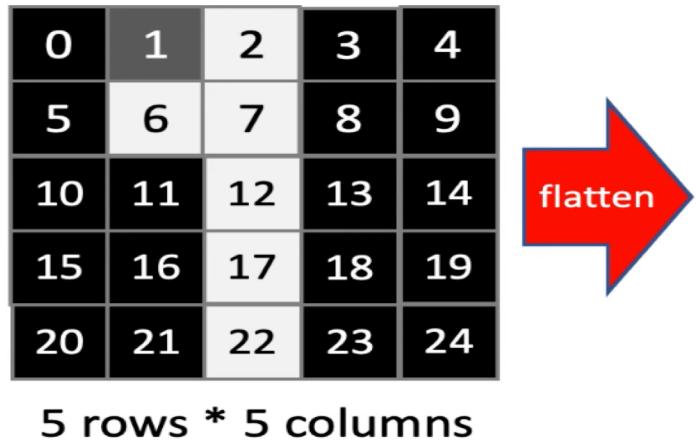
- MLPs' demerits

- Too many parameters

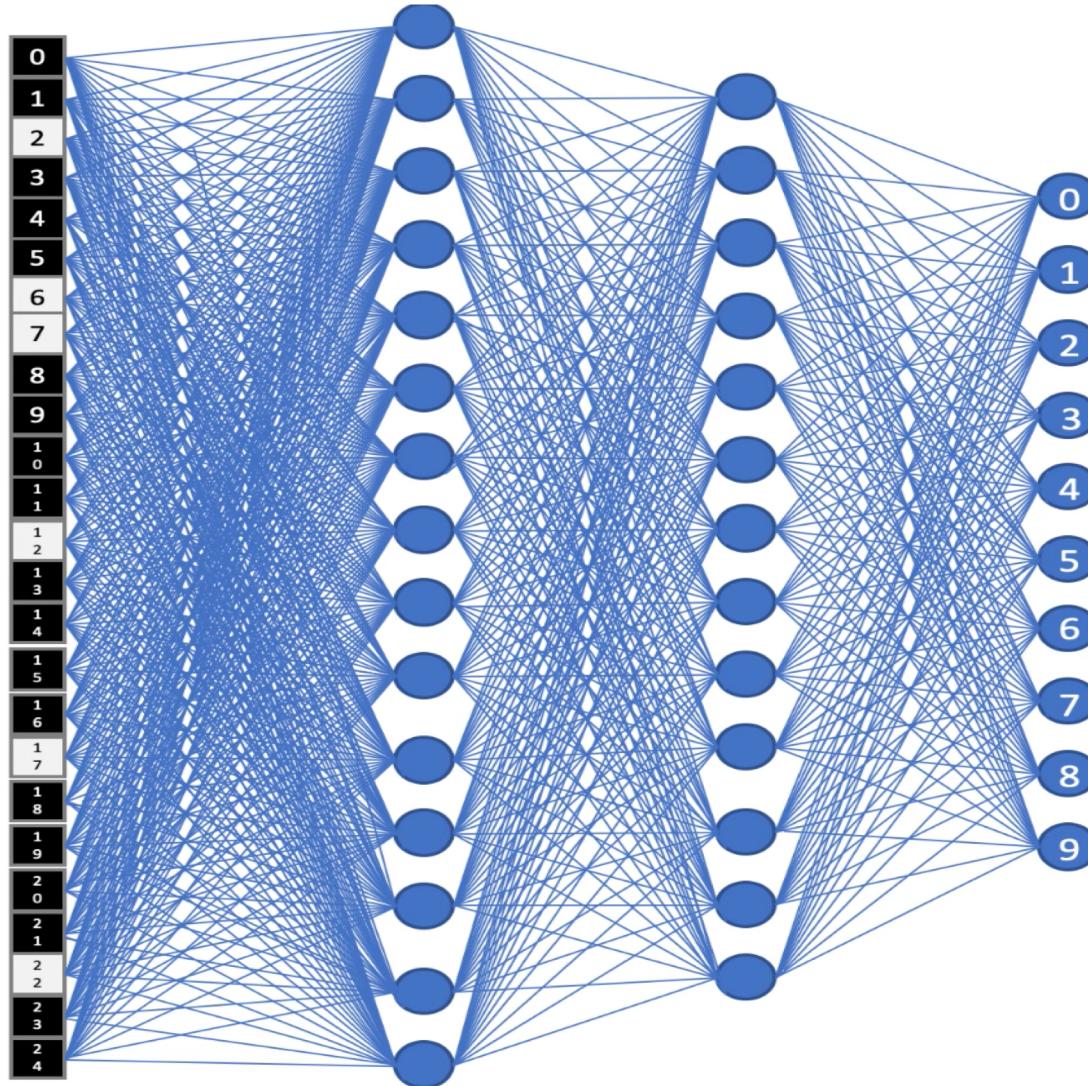
- Naïve



Review the key points



25 rows



Review the key points

0	1	2	3	4
5	6	7	8	9
10	11	12	13	14
15	16	17	18	19
20	21	22	23	24

0	1	2	3	4
5	6	7	8	9
10	11	12	13	14
15	16	17	18	19
20	21	22	23	24



Train data

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
---	---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Train data

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
---	---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Test data

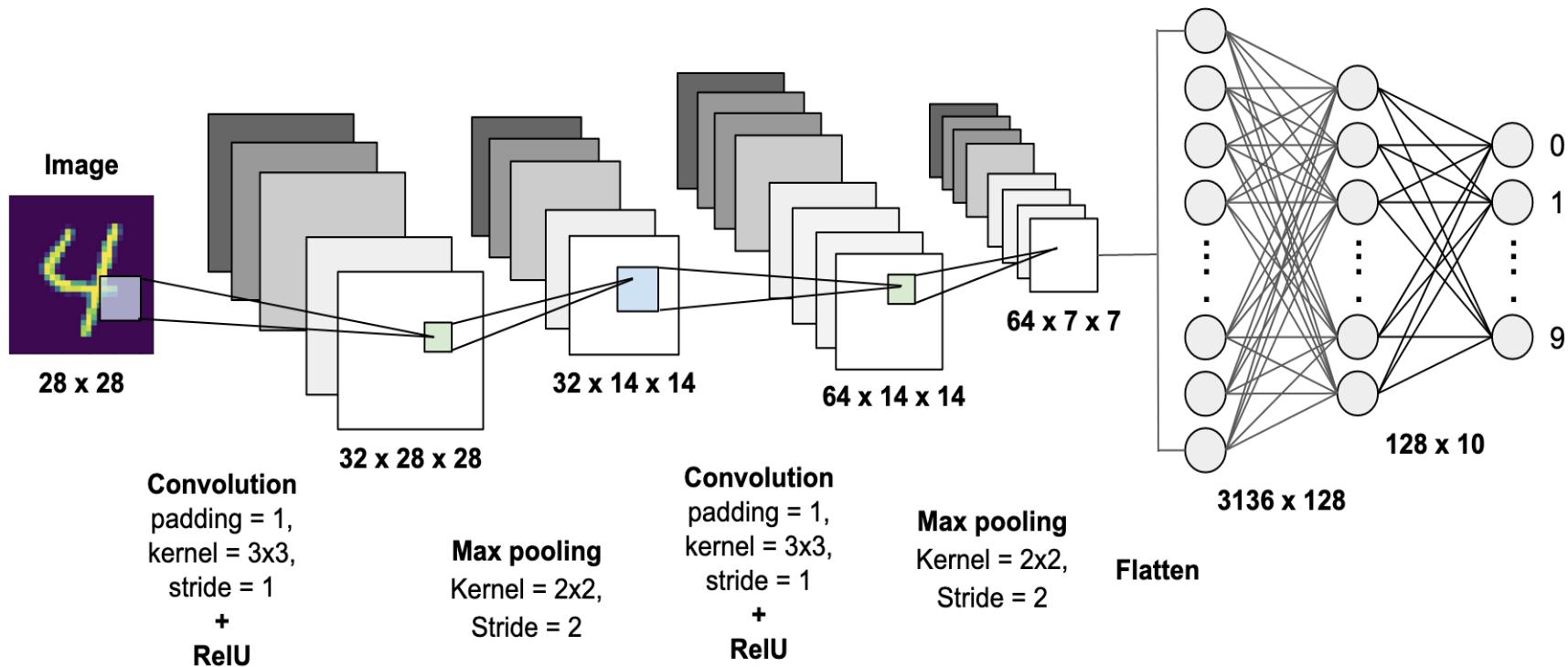
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
---	---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Test data

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
---	---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Review the key points

- Convolutional Neural Network



Convolution

Kernel
Filter

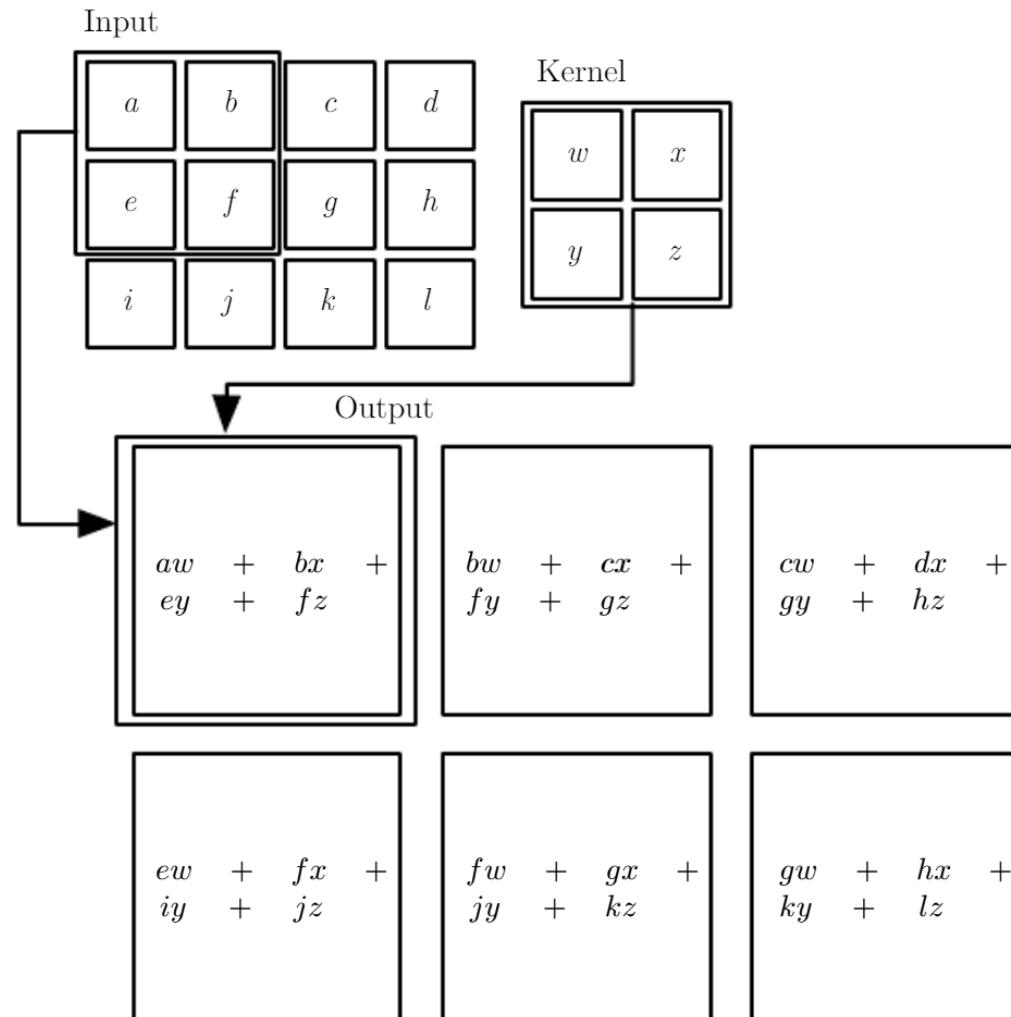
Pooling

Padding
Stride

Flatten
Dense

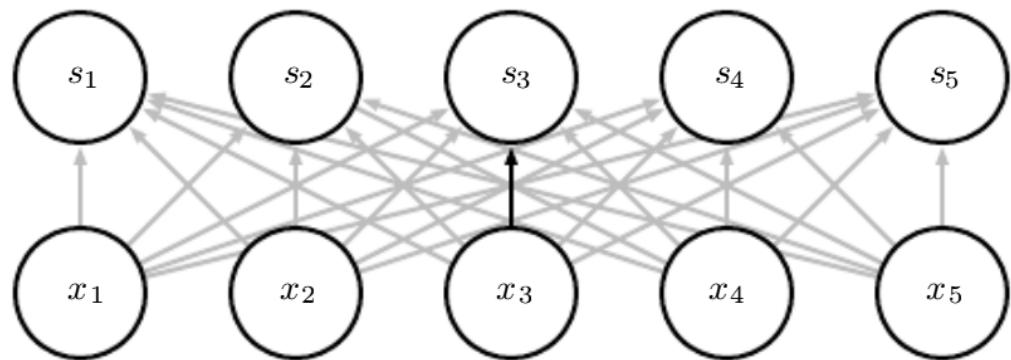
Review the key points

- Convolution

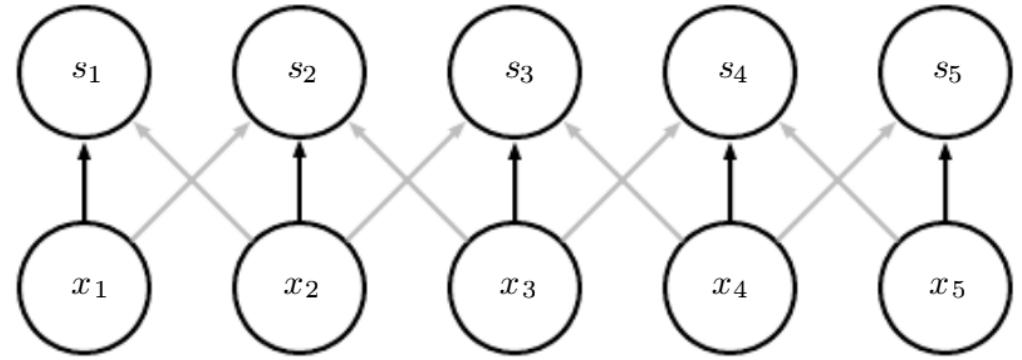


Review the key points

- Parameter sharing



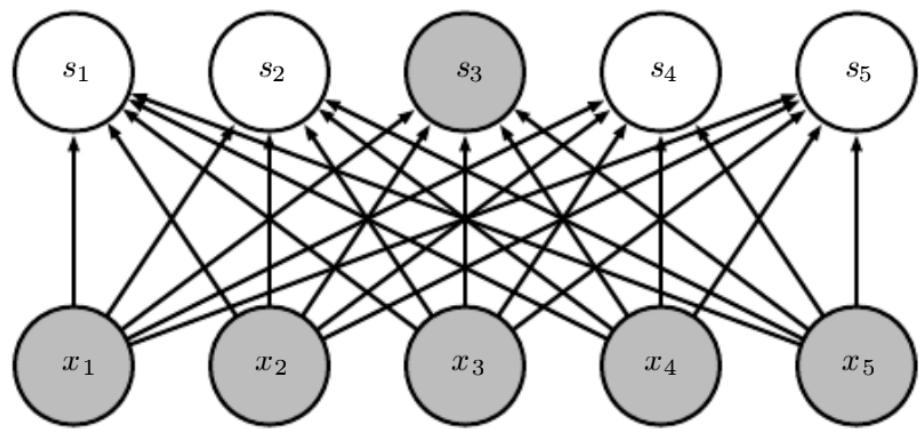
Fully connected layer



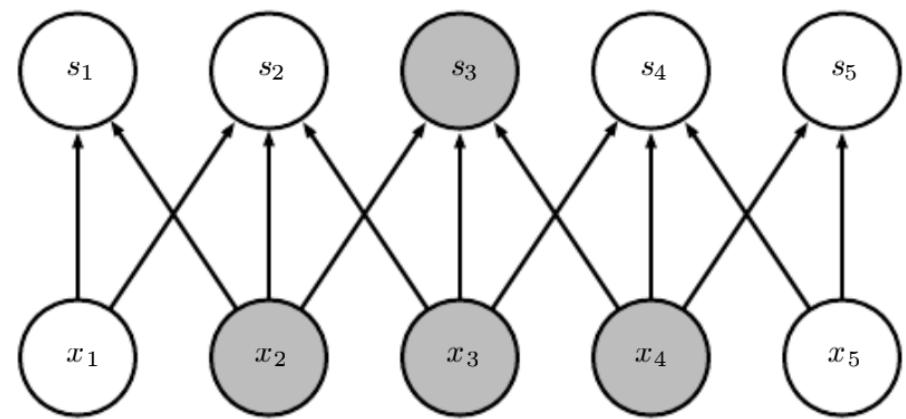
Convolution layer

Review the key points

- Sparse Interaction



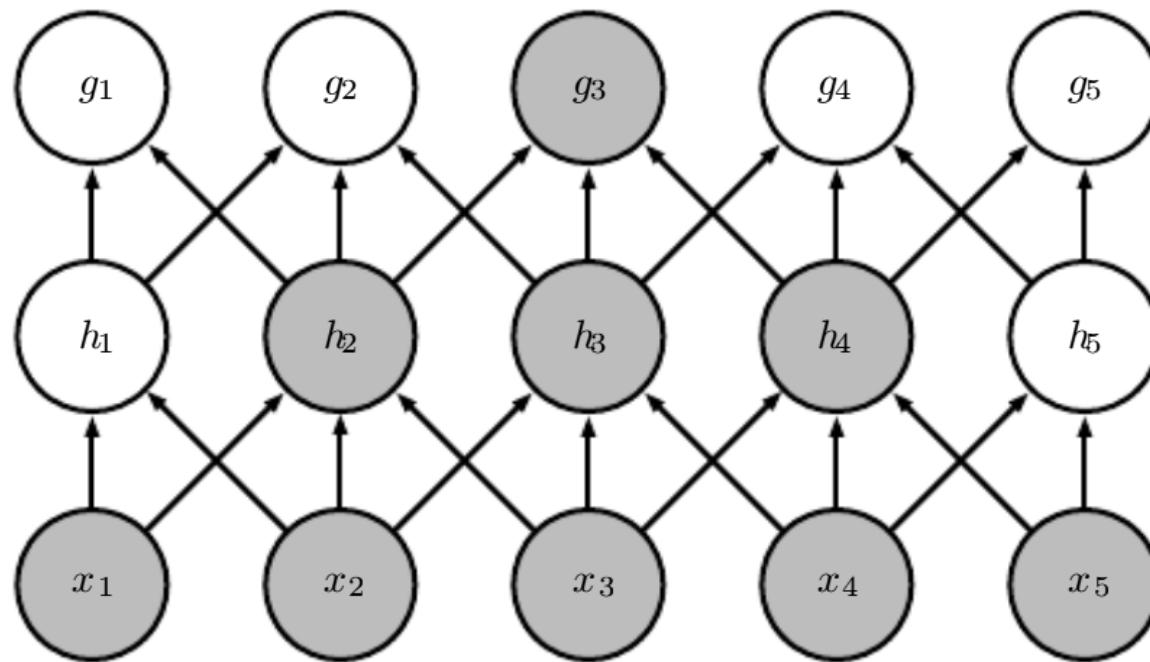
Fully connected layer



Convolution layer

Review the key points

- Sparse Interaction



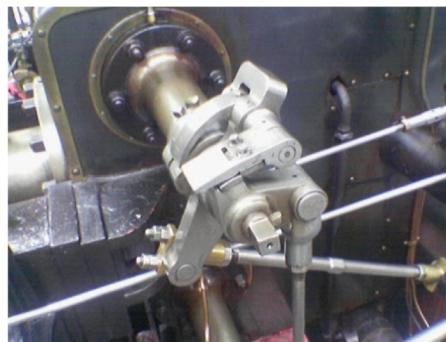
Review the key points

- Role of kernel

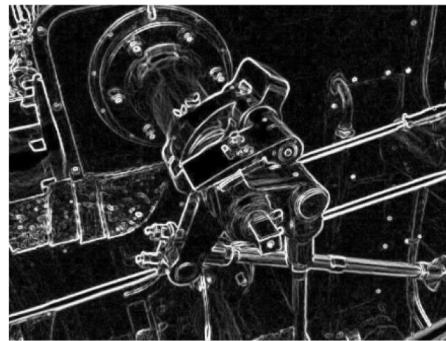
$$S_x = \begin{array}{|c|c|c|} \hline 1 & 0 & -1 \\ \hline 2 & 0 & -2 \\ \hline 1 & 0 & -1 \\ \hline \end{array} \quad S_y = \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 0 & 0 & 0 \\ \hline -1 & -2 & -1 \\ \hline \end{array}$$

What do they mean?

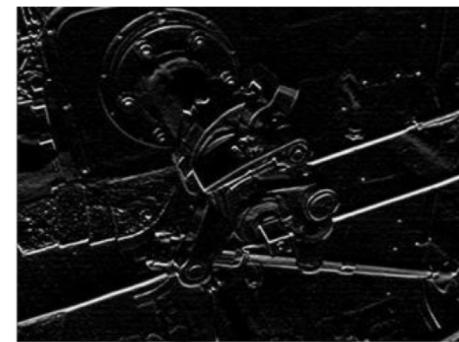
Original Image
 I



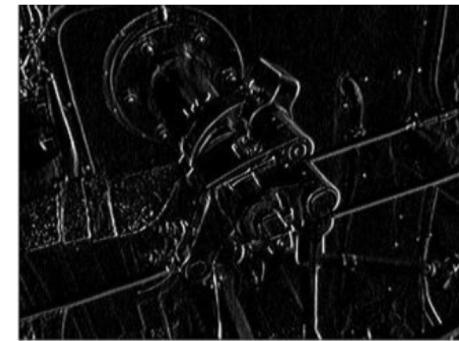
$|I * S_x| + |I * S_y|$



$I * S_y$

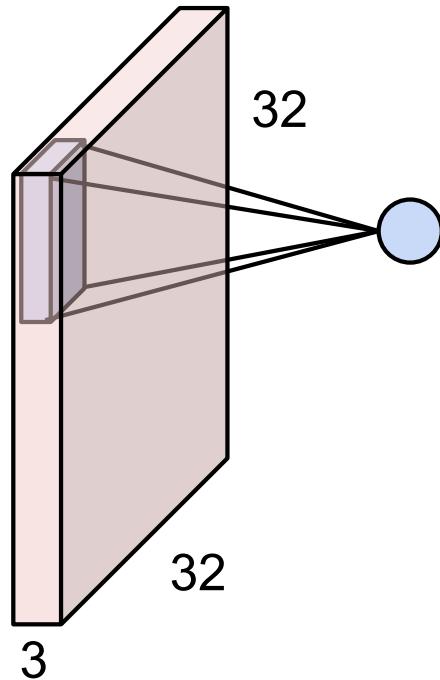


$I * S_x$



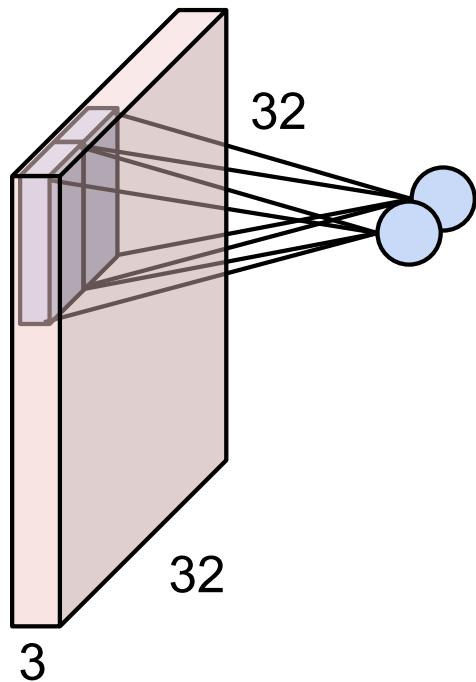
Review the key points

- Convolution



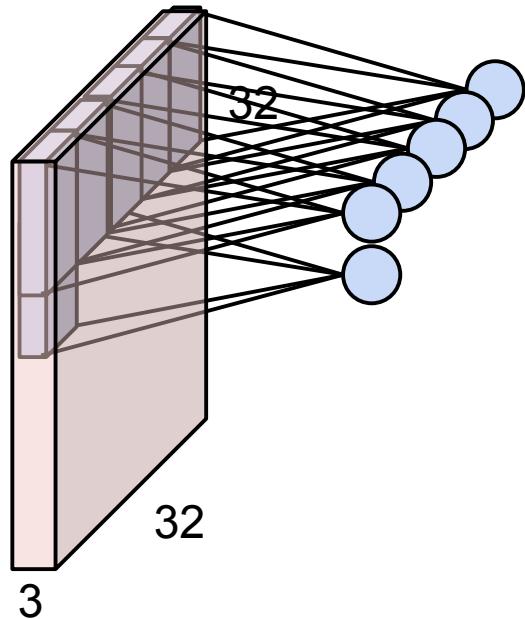
Review the key points

- Convolution



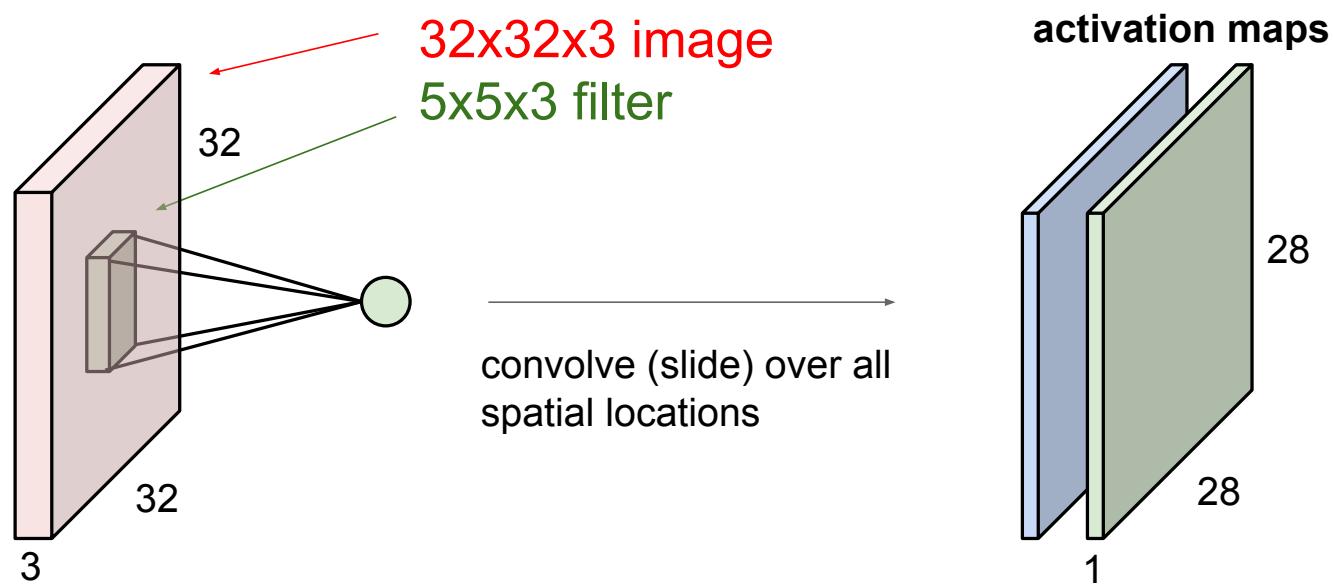
Review the key points

- Convolution



Review the key points

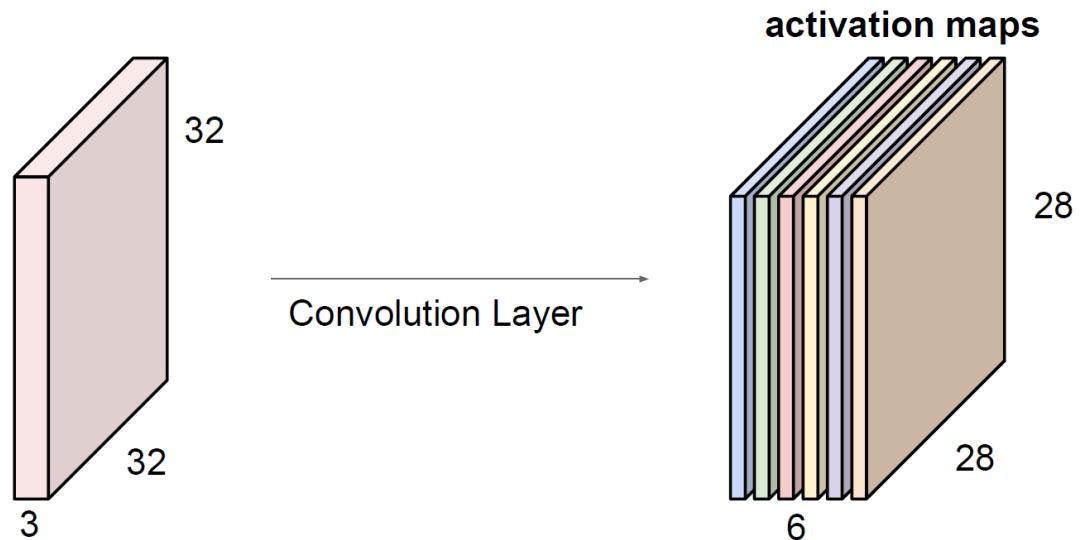
- Convolution



Review the key points

- Convolution

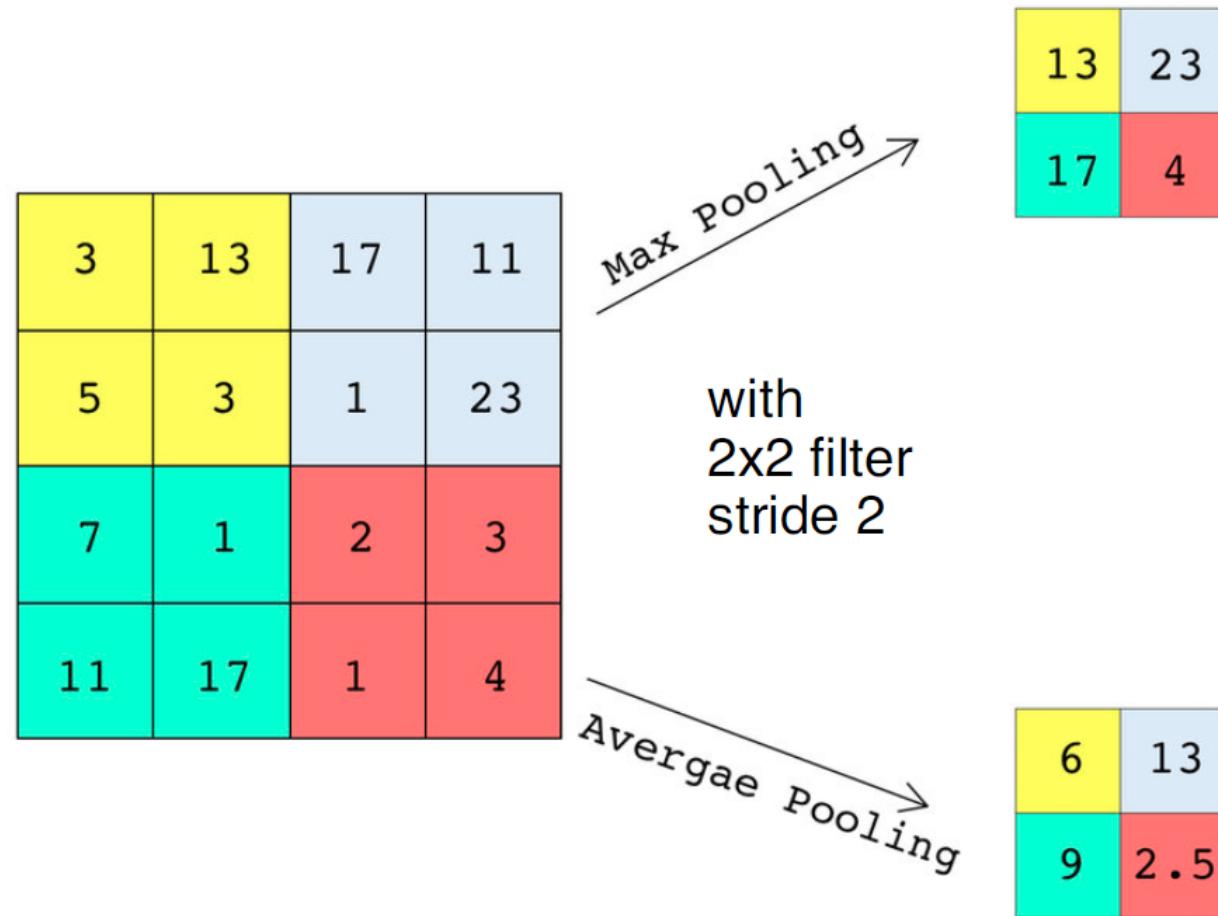
For example, if we have 6 5x5 filters, we will get 6 separate activation maps:



We stack these up to get a “new image” of size 28x28x6!

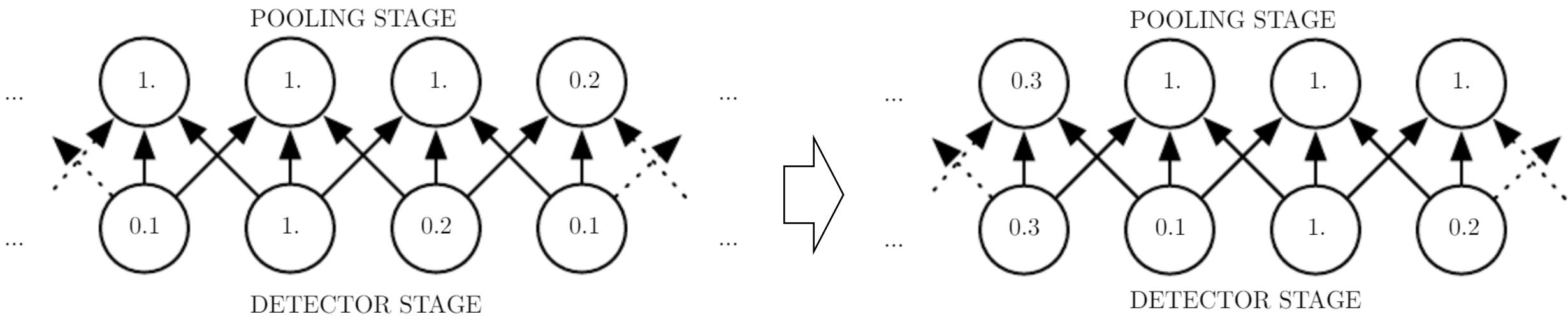
Review the key points

- Pooling
 - Max pooling
 - Average pooling



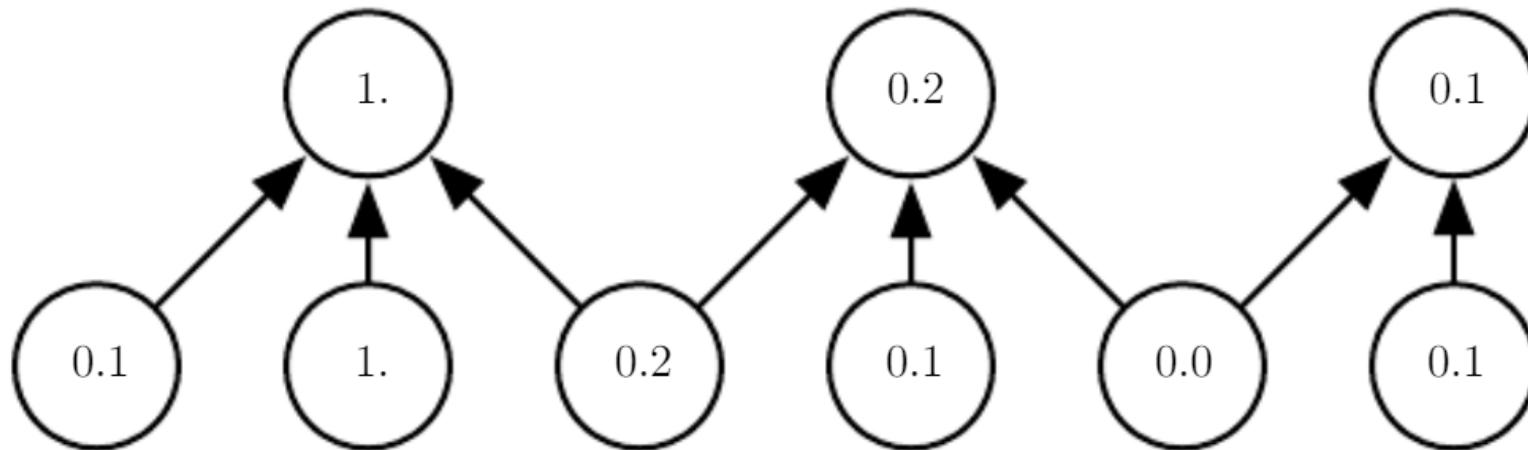
Review the key points

- Pooling의 효과
 - Invariance



Review the key points

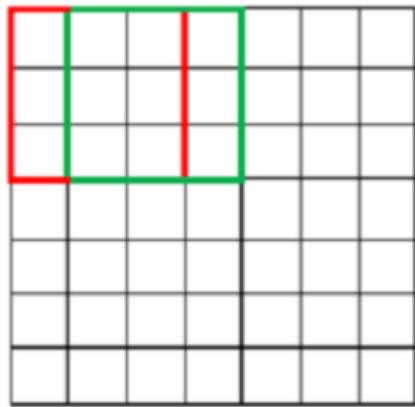
- Pooling의 효과
 - Summarizing :



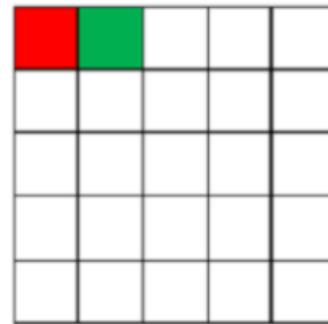
Review the key points

- Stride

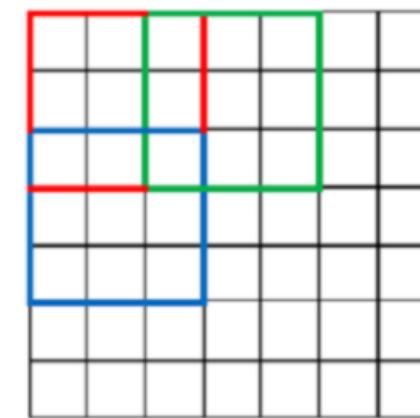
7 x 7 Input Volume



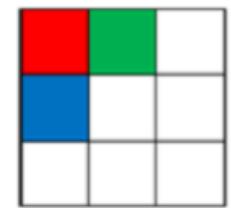
5 x 5 Output Volume



7 x 7 Input Volume



3 x 3 Output Volume



Stride with value 1 (default)

Stride with value 2

Review the key points

- Zero padding

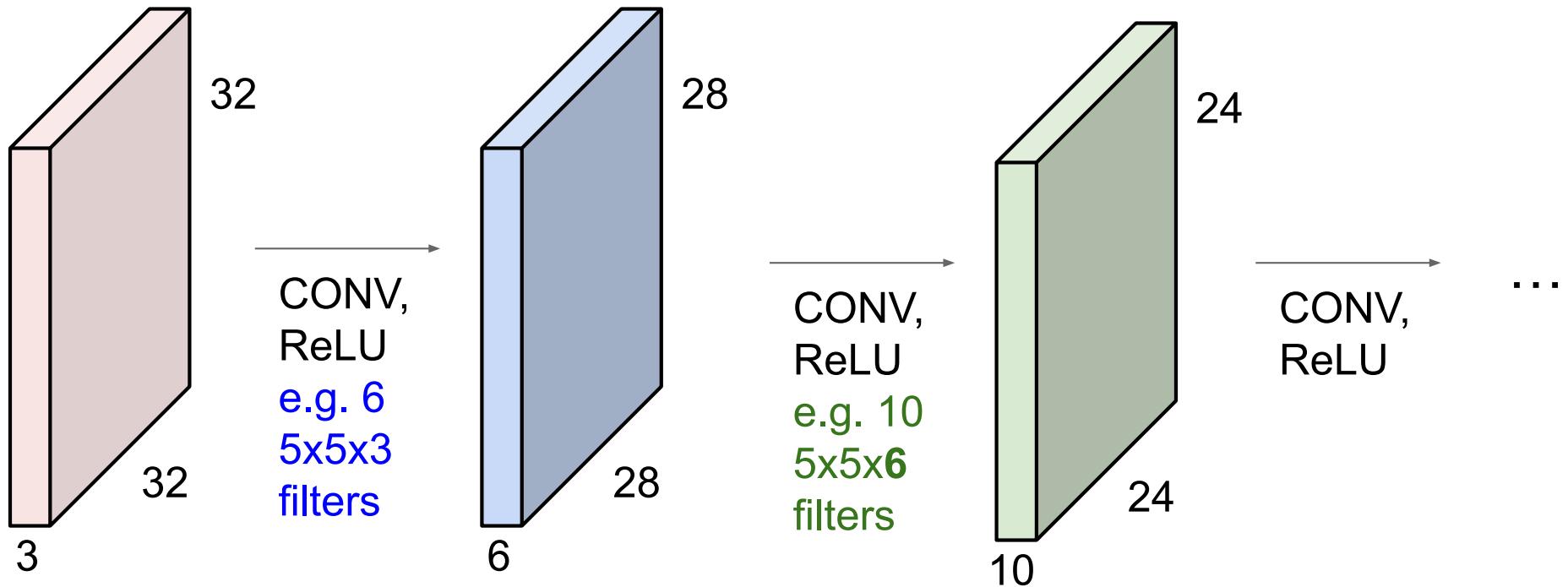
Keep the sizes

Reflect edges' information more

0	0	0	0	0	0		
0							
0							
0							
0							

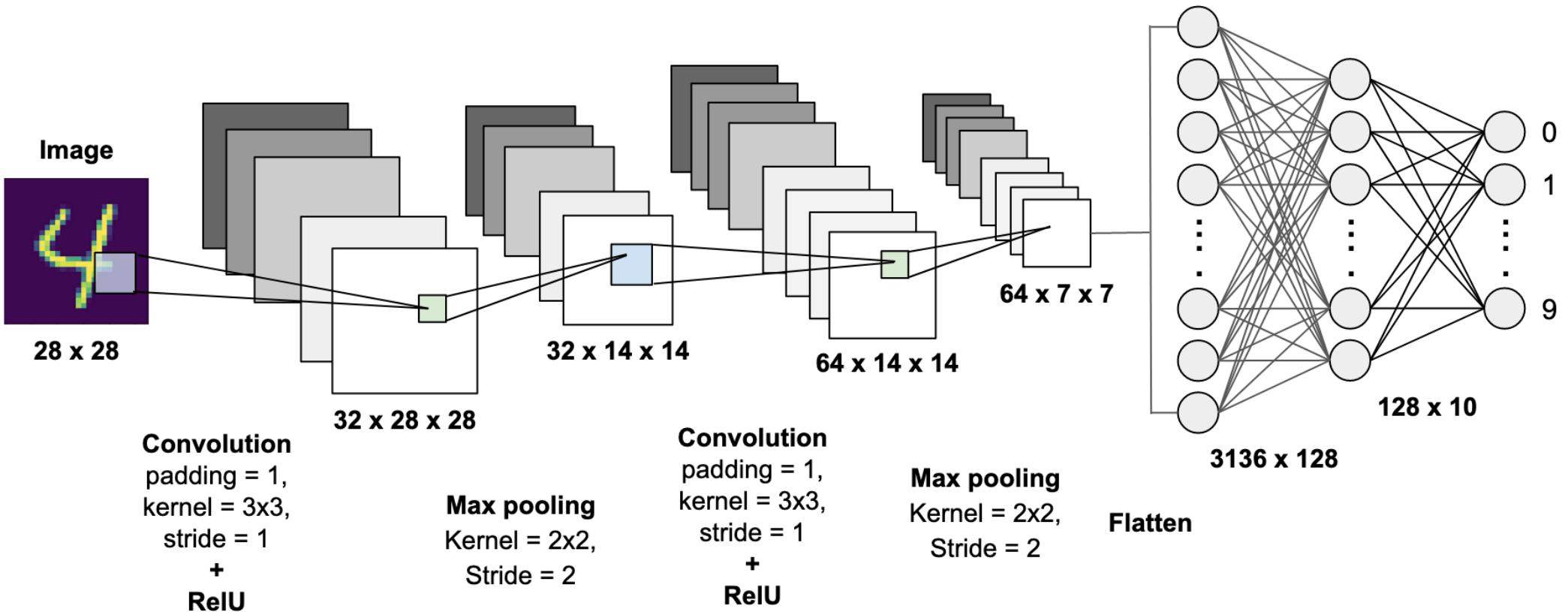
Review the Keypoints

- Size



Review the key points

- The number of parameters



Review the key points

- Output size of convolution

- <https://ezyang.github.io/convolution-visualizer/index.html>
- Input size: $W_1 \times H_1 \times D_1$
- Requires four hyper-parameters:
 - Number of filters K
 - Filter size F
 - Stride S
 - Zero padding P
- Produces an output of size $W_2 \times H_2 \times D_2$ where
 - $W_2 = (W_1 - F + 2P)/S + 1$, $H_2 = (H_1 - F + 2P)/S + 1$
 - $D_2 = K$

CNN - Summary

- CNNs
 - Are used for all aspects of computer vision, and have won numerous pattern recognition competitions
 - Able learn interpretable features at different levels of abstraction
 - Typically, consist of convolution layers, pooling layers, nonlinearities, and fully connected layers
- Other resources
 - <https://cs231n.github.io/convolutional-networks/>

Table of Contents

- CNN
 - Review the key points
 - Practice
- ResNet
 - Review the key points
 - Practice



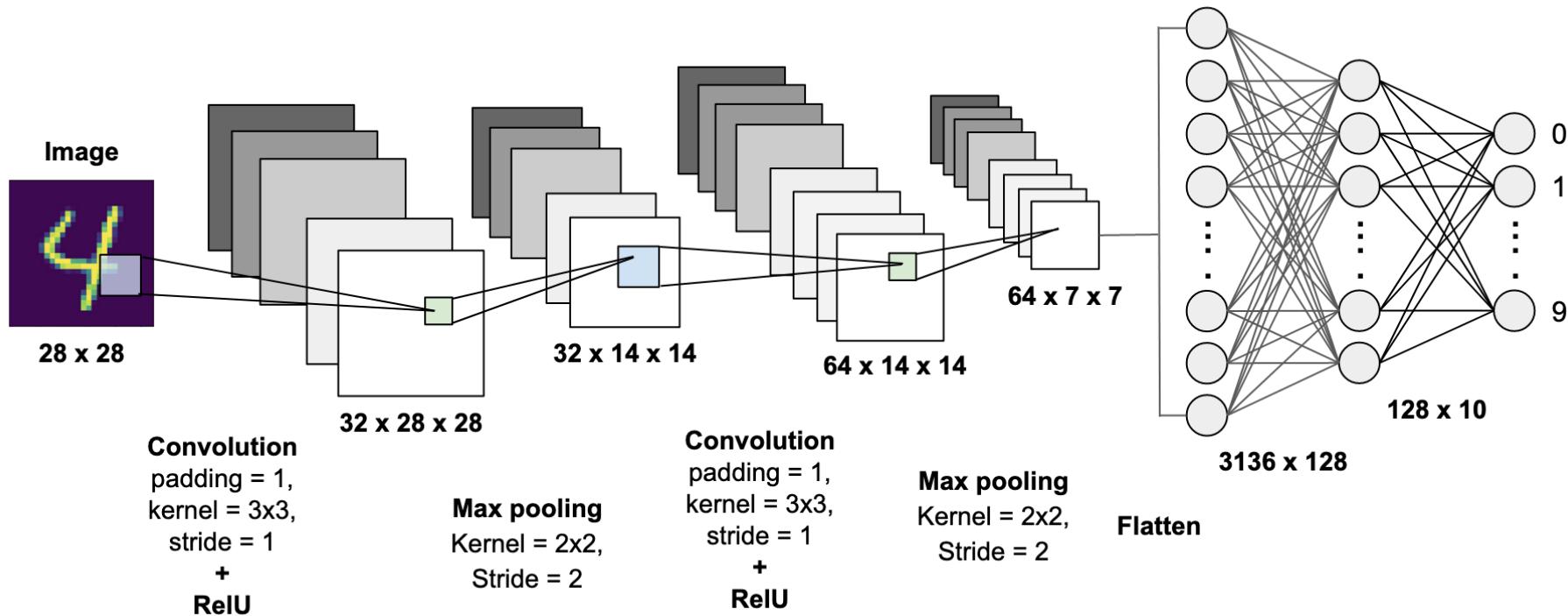
		<ul style="list-style-type: none"> 특징추출작업(Feature extraction)
1. Conv	Conv2D (input_shape= (28, 28, 1), kernel_size= (3,3) filters = 32 Strides = (1,1) Activation ='relu' use_bias =True, Padding =valid)	<ul style="list-style-type: none"> Input_shape = Tensor, (높이, 너비, 채널) 커널값이 하나인 경우 가로 세로 같음 Padding = 'valid' 또는 'same'
2. Pool	MaxPool2D (pool_size=(2,2) , padding =valid)	<ul style="list-style-type: none"> 이 경우 2x2 단위로 pooling 수행
3. Conv	Conv2D (kernel_size(3,3) Filters =32 Strides = Activation =	<ul style="list-style-type: none"> Input값 자동 적용



5. Flat	Flatten()	<ul style="list-style-type: none"> 분류작업 시작(Classification) MLP 적용 위해 Tensor 를 1차원으로 나열
6. Dense	Dense (256, activation='relu') Dense(256, activation='relu') Dense(10, activation ='softmax')	<ul style="list-style-type: none"> MLP에서 FC 와 완전한 동일
7. Dropout	Dropout(0.4)	<ul style="list-style-type: none"> Overfitting을 막기 위함

Practice

- 해당 구조를 가지는 CNN을 구현하고 MNIST 데이터 분류해보기



Reference

- Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016).
<http://www.deeplearningbook.org>

Thank You :)

Slee2020@postech.ac.kr