deeplearning.ai

Convolutional Neural Networks

Computer vision

+ 음성인식에도 활용

Computer Vision Problems

Image Classification

64x64

→ Cat? (0/1)

Neural Style Transfer

Object detection

차의 여부와 위치
박스로 식별

Andrew Ng

Deep Learning on large images

input이 크다!

64x64 × 3

12288

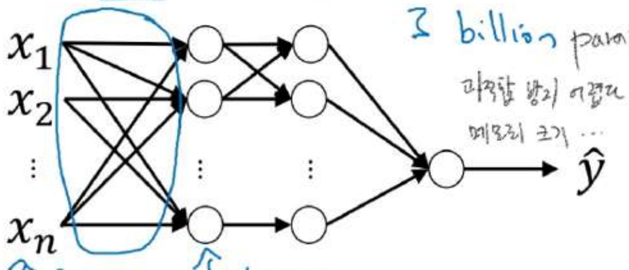
→ Cat? (0/1)

RGB 채널

1000×1000×3
= 3 million

$x \in \mathbb{R}^{3M}$

$W^{[1]}$ (1000, 3M)



3M

1000

3 billion parameter

과적합 방지 어렵다
메모리 크기 ...

Andrew Ng

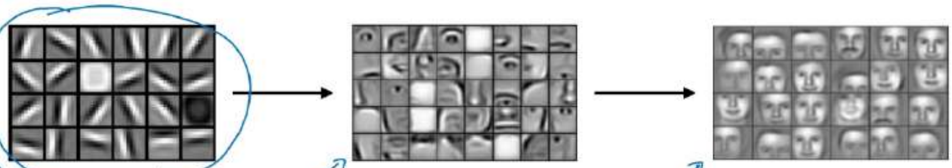
Convolutional Neural Networks

deeplearning.ai

Edge detection example

"합성곱 작업"

Computer Vision Problem



detect edges!

vertical edges

수직 모서리 찾기

수평 〃

horizontal edges

Andrew Ng

Vertical edge detection

$3\times1 + 1\times1 + 2\times1 + 0\times0 + 5\times0 + 7\times0 + 1\times-1 + 8\times-1 + 2\times-1 = -5$

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6×6

"convolution"

*

1	0	-1
1	0	-1
1	0	-1

3×3

filter (kernel)

=

-5	-4	0	8
-10	-2	2	3
0	-2	-4	-7
-3	-2	-3	-16

4×4

python : Conv Forward 함수

* 대신

tensorflow : tf.nn.conv2d

keras : Conv2D

Vertical edge detection

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

6×6

*

밝은 픽셀 어두운 픽셀

1	0	-1
1	0	-1
1	0	-1

3×3

=

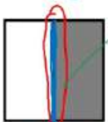
0	30	30	0
0	30	30	0
0	30	30	0
0	30	30	0

4×4

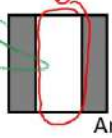
Q

-1 과 1 위치를 바꾼다면?

경계선 찾음!



*



Andrew Ng

deeplearning.ai

Convolutional Neural Networks

More edge detection

Vertical edge detection examples

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

*

1	0	-1
1	0	-1
1	0	-1

=

0	30	30	0
0	30	30	0
0	30	30	0
0	30	30	0

밝 → 어

0	0	0	10	10	10
0	0	0	10	10	10
0	0	0	10	10	10
0	0	0	10	10	10
0	0	0	10	10	10
0	0	0	10	10	10

*

1	0	-1
1	0	-1
1	0	-1

=

0	-30	-30	0
0	-30	-30	0
0	-30	-30	0
0	-30	-30	0

어 → 밝

Andrew Ng

Vertical and Horizontal Edge Detection

1	0	-1
1	0	-1
1	0	-1

Vertical

1	1	1
0	0	0
-1	-1	-1

Horizontal

… 유사

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
0	0	0	10	10	10
0	0	0	10	10	10
0	0	0	10	10	10

6x6

*

1	1	1
0	0	0
-1	-1	-1

=

0	0	0	0
30	10	-10	-30
30	10	-10	-30
0	0	0	0

Andrew Ng

Learning to detect edges

1	0	-1
1	0	-1
1	0	-1

1	0	-1
2	0	-2
1	0	-1

중간 강조 →

Sobel filter

3	0	-3
10	0	-10
3	0	-3

Scharr filter

중요!

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

convolution

*

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

$\underbrace{\qquad}_{3\times 3}$

역전파로 학습

→ 더 좋은 필터 학습

=

45°
70°
73° 어떤 형태의 윤곽도 학습

Andrew Ng

Convolutional Neural Networks

Padding

deeplearning.ai

Padding

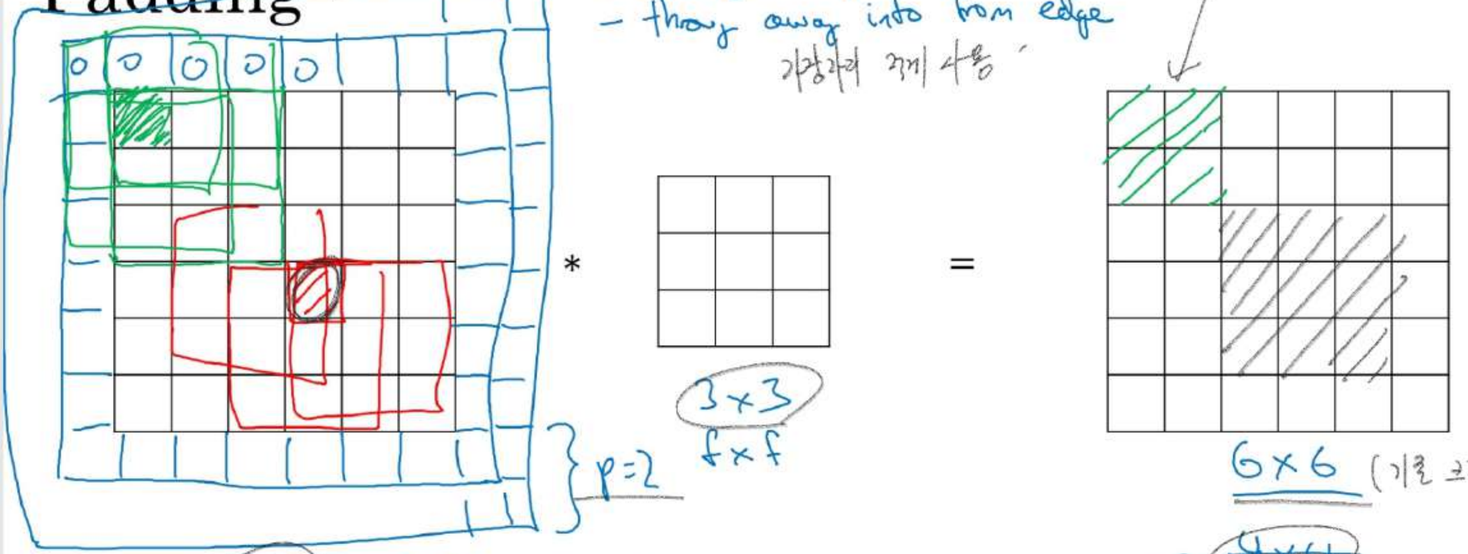
해결 이미지 덧대기 ⟨2단점⟩

이미지 축소

- shrinking output
- throw away info from edge

가장자리 정보 사용

가장자리 정보를 덜 가져오는 것을 줄인다!



$6\times6 \rightarrow 8\times8$

$n\times n$

$n-f+1 \times n-f+1$

$6-3+1=4$

$n+2p-f+1 \times n+2p-f+1$

$6+2-3+1 \times \_\_ = 6\times6$

p = padding = $\cancel{1}$ 2

패딩의 양

Andrew Ng

Valid and Same convolutions

패딩을 얼만큼 할 건인가

→ no padding

"Valid": $n\times n \; * \; f\times f \rightarrow n-f+1 \times n-f+1$

유효 합성곱 $6\times6 \; * \; 3\times3 \rightarrow 4\times4$

"Same": Pad so that output size is the same as the input size.

동일 합성곱

$n+2p-f+1 \times n+2p-f+1$

$\cancel{n}+2p-f+1=\cancel{n} \Rightarrow \boxed{p=\frac{f-1}{2}}$

ex $3\times3 \quad p=\frac{3-1}{2}=1$

ex $5\times5 \quad f=5 \quad p=2$

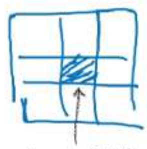
f 거의 항상 홀수 usually odd

1×1 없음

3×3

5×5

7×7



홀수는 중심이 존재!

Andrew Ng

Convolutional
Neural Networks

deeplearning.ai

Strided
convolutions

Strided convolution

2^{3}	3^{4}	7^{3}	4^{4}	6^{3}	2^{4}	9^{4}
6^{1}	6^{0}	9^{1}	8^{0}	7^{1}	4^{0}	3^{2}
3^{3}	4^{4}	8^{3}	3^{4}	8^{3}	9^{4}	7^{4}
7^{1}	8^{0}	3^{1}	6^{0}	6^{1}	3^{0}	4^{2}
4^{3}	2^{4}	1^{3}	8^{4}	3^{3}	4^{4}	6^{4}
3^{1}	2^{0}	4^{1}	1^{0}	9^{1}	8^{0}	3^{2}
0^{-1}	1^{0}	3^{-1}	9^{0}	2^{-1}	1^{0}	4^{3}

$\underline{7\times7}$

*

3	4	4
1	0	2
-1	0	3

$\underline{3\times3}$

=

91	100	83
69	91	127
44	72	74

$\underline{3\times3}$

$\underline{\text{stride} = 2}$

$\lfloor z \rfloor = \text{floor}(z)$

$n \times n \quad * \quad f \times f$

padding p stride s

$s = 2$

$$\left\lfloor \frac{n+2p-f}{s} + 1 \right\rfloor \times \left\lfloor \frac{n+2p-f}{s} + 1 \right\rfloor$$

정수가 아닐 시 내림을 한다.

$$\frac{7+0-3}{2} + 1 = \frac{4}{2} + 1 = 3$$

Andrew Ng

Summary of convolutions

$n \times n$ image $f \times f$ filter

padding p stride s

Output Size:

$$\left\lfloor \frac{n+2p-f}{s} + 1 \right\rfloor \times \left\lfloor \frac{n+2p-f}{s} + 1 \right\rfloor$$

Technical note on cross-correlation vs. convolution

교차상관

Convolution in math textbook:

2^{7}	3^{2}	7^{5}	4	6	2
6^{9}	6^{0}	9^{4}	8	7	4
3^{-1}	4^{1}	8^{3}	3	8	9
7	8	3	6	6	3
4	2	1	8	3	4
3	2	4	1	9	8

*

3	4	5
1	0	2
-1	9	7

=

7	2	5
9	0	4
-1	1	3

미러링 과정 생략!!

결합 법칙

$(A * B) * C = A * (B * C)$

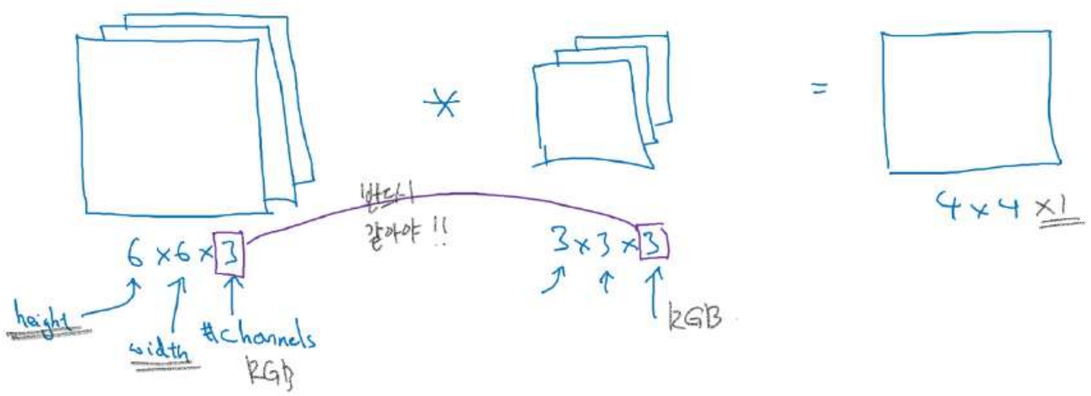
Andrew Ng

deeplearning.ai

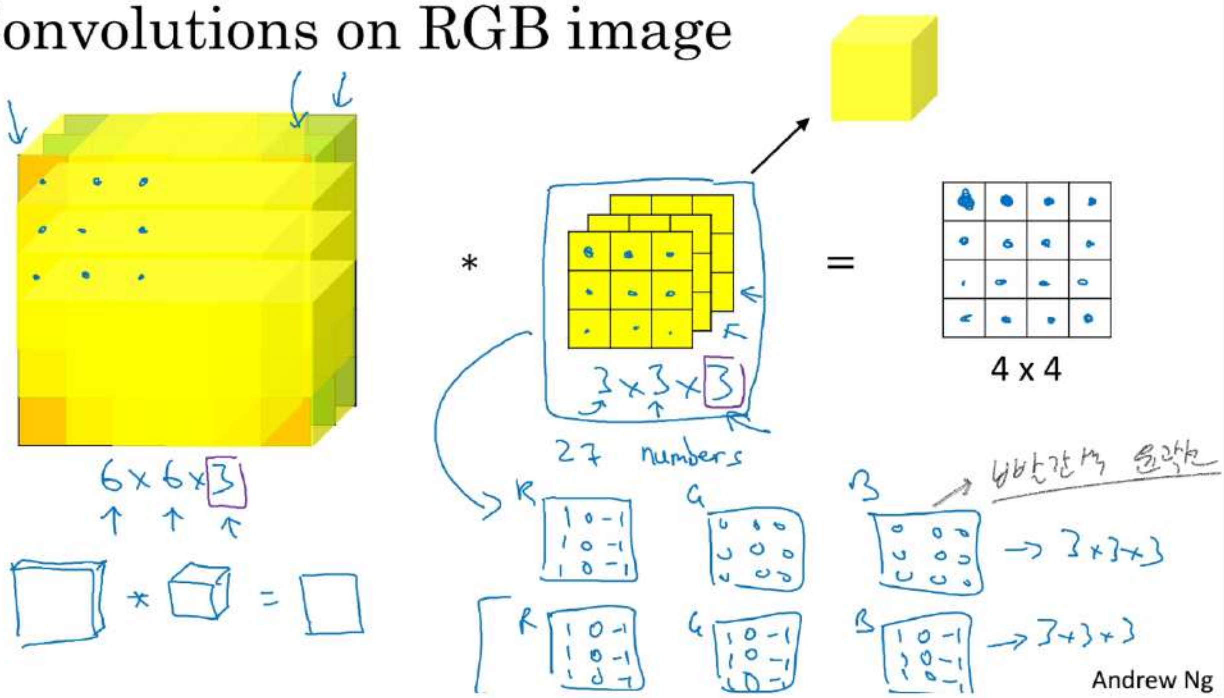
Convolutional
Neural Networks

Convolutions over
volumes

Convolutions on RGB images

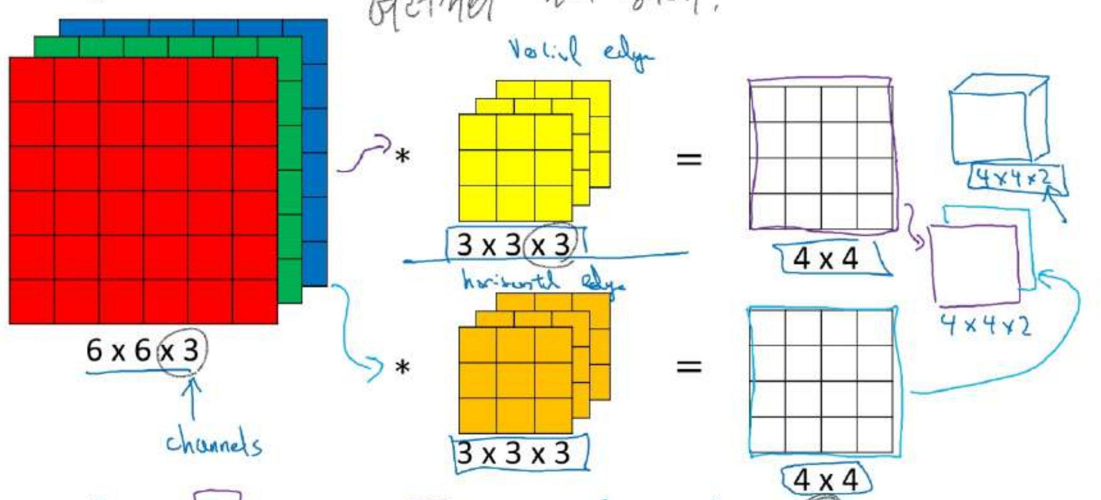


Convolutions on RGB image



Multiple filters

여러개의 필터 동시에!



Summary: $n \times n \times n_c \ * \ f \times f \times n_c \rightarrow n-f+1 \times n-f+1 \times n_c'$

$6 \times 6 \times 3 \qquad 3 \times 3 \times 3 \qquad 4 \times 4 \times 2$ ← #filters

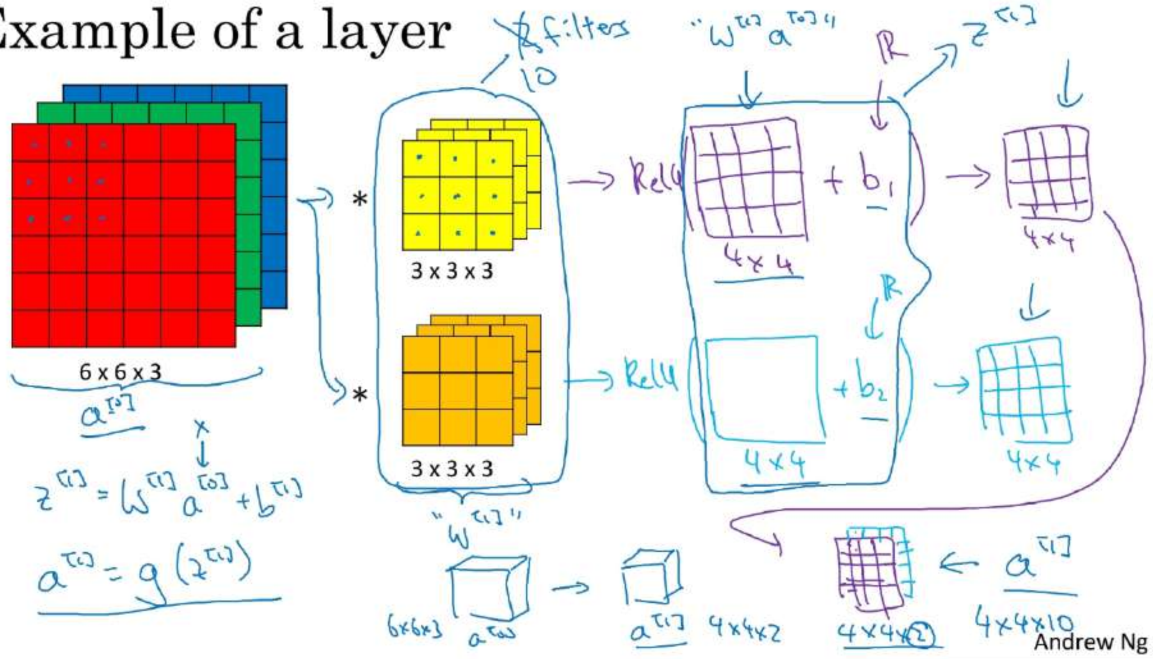
Andrew Ng

deeplearning.ai

Convolutional Neural Networks

One layer of a convolutional network

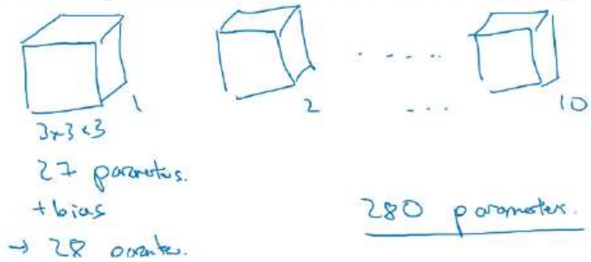
Example of a layer



Andrew Ng

Number of parameters in one layer

If you have 10 filters that are 3 x 3 x 3 in one layer of a neural network, how many parameters does that layer have?



$n_H \times n_W \times n_c^{[l-1]} \rightarrow$ " $[l]$

$n_H^{[l]} = \frac{n_H^{[l-1]} + 2p^{[l]} - f^{[l]}}{s^{[l]}} + 1$

$n_W^{[l]} = \frac{n_W^{[l-1]} + 2p^{[l]} - f^{[l]}}{s^{[l]}} + 1$

Summary of notation

If layer l is a convolution layer:

$f^{[l]}$ = filter size

$p^{[l]}$ = padding

$s^{[l]}$ = stride

$\boxed{n_c^{[l]}}$ = number of filters

→ Each filter is: $f^{[l]} \times f^{[l]} \times \boxed{n_c^{[l-1]}}$

Activations: $a^{[l]} \rightarrow n_H^{[l]} \times n_W^{[l]} \times n_c^{[l]}$

Weights: $f^{[l]} \times f^{[l]} \times n_c^{[l-1]} \times n_c^{[l]}$

bias: $n_c^{[l]}$ — $(1, 1, 1, n_c^{[l]})$ ← #filters in layer l.

Input: $\boxed{n_H^{[l-1]} \times n_W^{[l-1]} \times \boxed{n_c^{[l-1]}}}$ ←

Output: $n_H^{[l]} \times n_W^{[l]} \times \boxed{n_c^{[l]}}$ ←

$n_{H/W}^{[l]} = \left\lfloor \frac{n_{H/W}^{[l-1]} + 2p^{[l]} - f^{[l]}}{s^{[l]}} + 1 \right\rfloor$

$A^{[l]} \rightarrow m \times \underbrace{n_H^{[l]} \times n_W^{[l]} \times n_c^{[l]}}$

$n_c \times n_H \times n_W$

Andrew Ng

deeplearning.ai

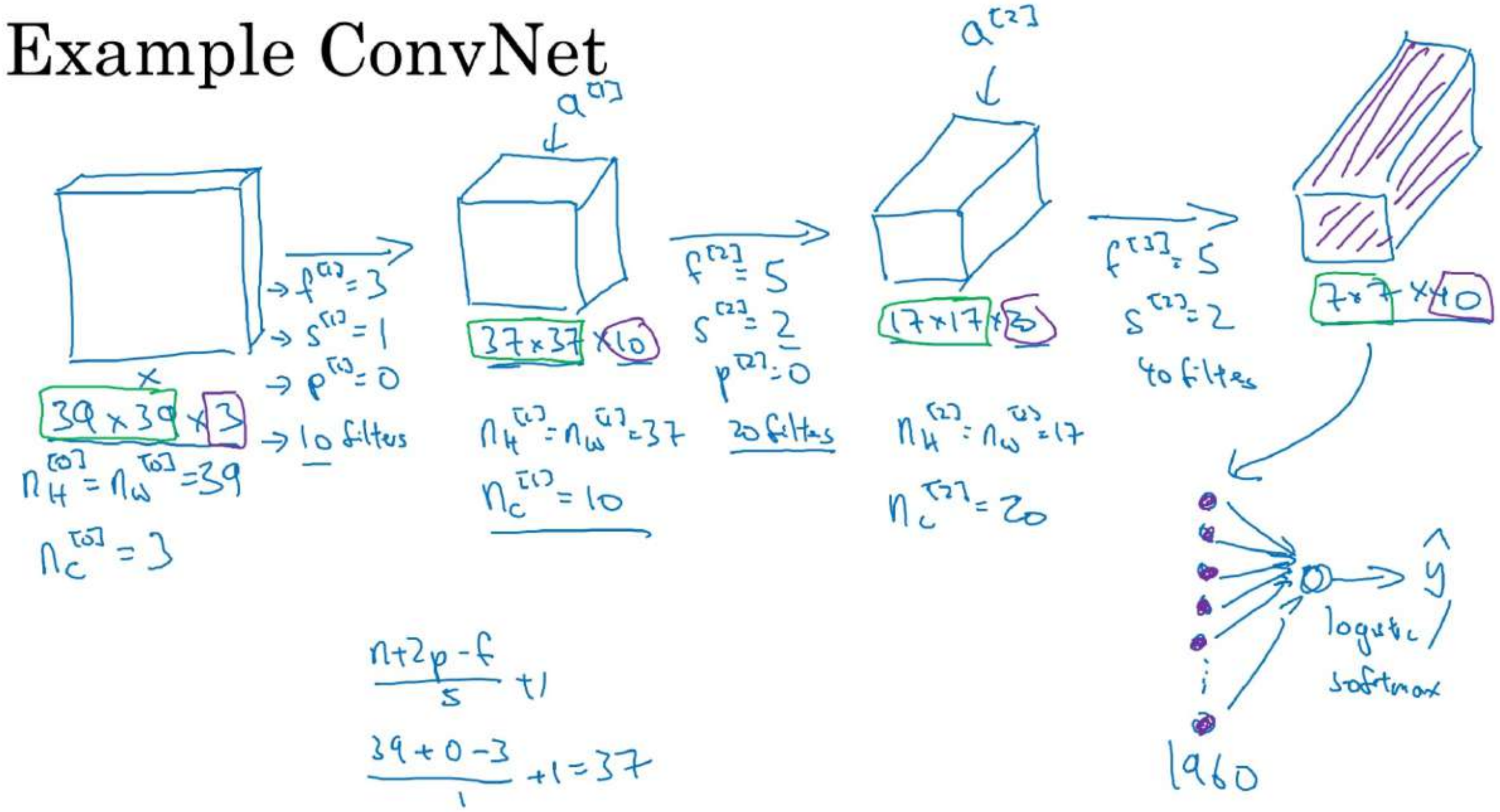
Convolutional Neural Networks

A simple convolution network example

합성곱 연산 → 편향 추가 → 활성화 함수

합성곱 신경망의 크기는 깊어질수록 줄어든다!

Example ConvNet



Types of layer in a convolutional network:

- Convolution (CONV)
- Pooling (POOL)
- Fully connected (FC)

deeplearning.ai

Convolutional Neural Networks

Pooling layers

Pooling layer: Max pooling

1	3	2	1
2	9	1	1
1	3	2	3
5	6	1	2

Pooling layer: Max pooling

1	3	2	1	3
2	9	[illegible]	1	5
1	[illegible]	[illegible]	[illegible]	2
8	3	[illegible]	1	0
5	6	1	2	9

$5 \times 5 \times \cancel{1}\ n_c$

9	9	5
9	9	5
8	6	9

$\boxed{\begin{matrix} f=3 \\ s=1 \end{matrix}}$

$3 \times 3 \times \cancel{1}\ n_c$

$\left\lfloor \frac{n+2p-f}{s} + 1 \right\rfloor$

Pooling layer: Average pooling

$4 \times 4 \quad \left(\frac{4+2\times 0-2}{2} + 1 \right) \rightarrow 2 \times 2$

1	3	2	1
2	9	1	1
1	4	2	3
5	6	1	2

$\rightarrow$

3.75	1.25
4	2

$f=2$

$s=2$

$7 \times 7 \times 1000 \rightarrow 1 \times 1 \times 1000$

Summary of pooling

Hyperparameters:

f : filter size

s : stride

Max or average pooling

$f=2, s=2$

$f=3, s=2$

~~p: padding~~

No parameters to learn!

$n_H \times n_W \times n_c$

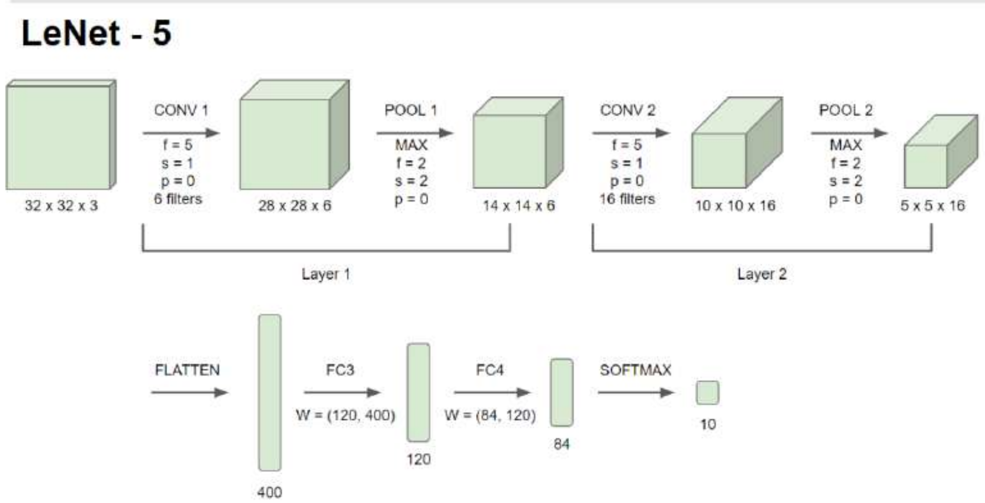
$\downarrow$

$\left\lfloor \frac{n_H-f}{s} + 1 \right\rfloor \times \left\lfloor \frac{n_W-f}{s} + 1 \right\rfloor \times n_c$

deeplearning.ai

Convolutional Neural Networks

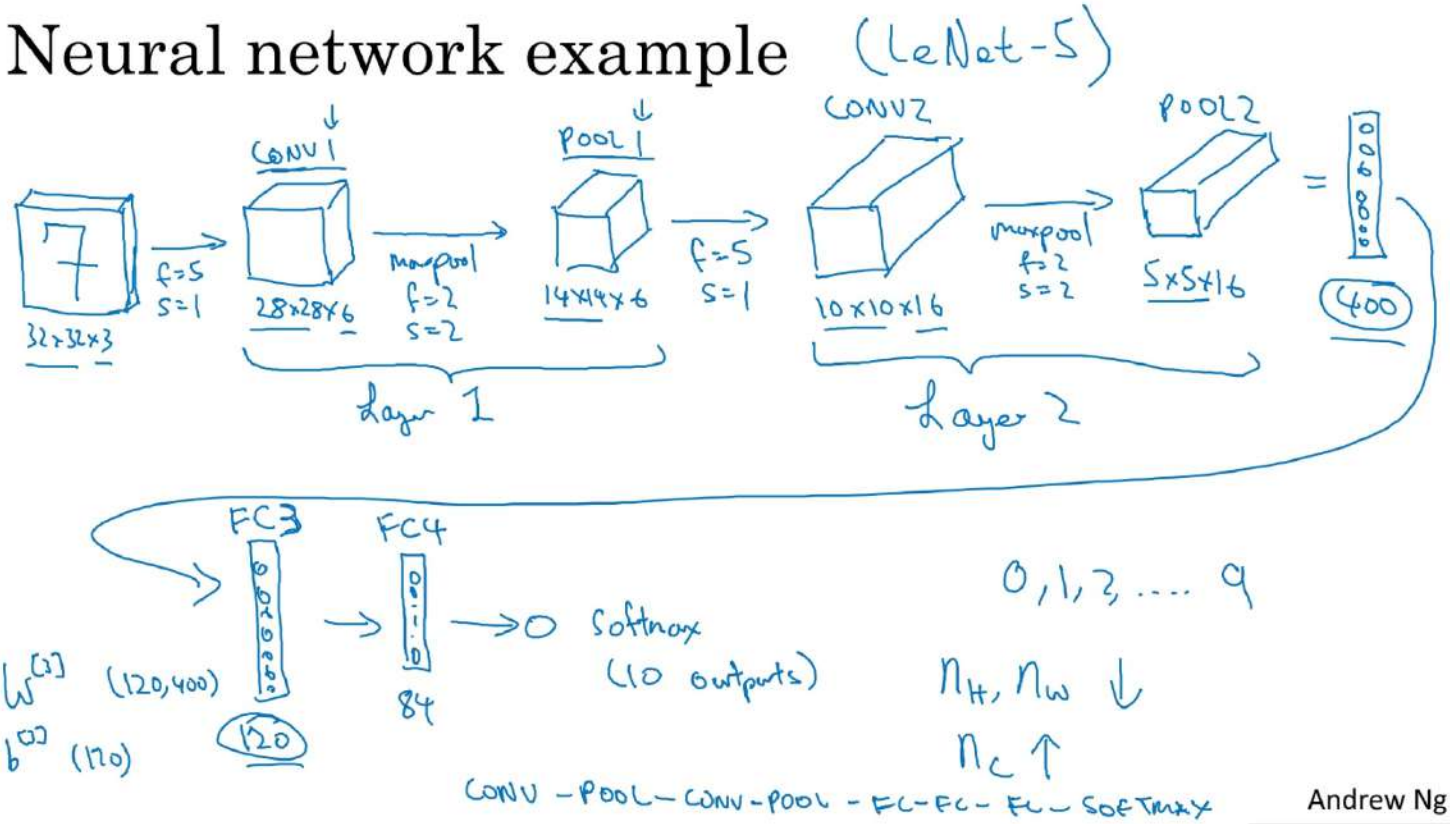
Convolutional neural network example



합성곱층, 풀링층 — 하나의 층으로

각각의 층으로

Neural network example (LeNet-5)



Andrew Ng

Neural network example

	Activation shape	Activation Size	# parameters
Input:	(32,32,3)	3,072 $a^{[0]}$	0
CONV1 (f=5, s=1)	(28,28,8)	6,272	608
POOL1	(14,14,8)	1,568	0
CONV2 (f=5, s=1)	(10,10,16)	1,600	3216
POOL2	(5,5,16)	400	0
FC3	(120,1)	120	48120
FC4	(84,1)	84	10164
Softmax	(10,1)	10	850

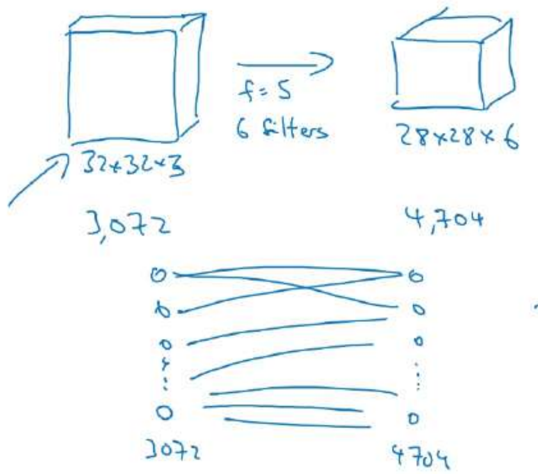
Andrew Ng

deeplearning.ai

Convolutional Neural Networks

Why convolutions?

Why convolutions



$5\times5 - 25$

26

$6\times26 = 156$ parameters

$3{,}072 \times 4{,}704 \approx 14M$

Why convolutions

translation invariance

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

$*$

1	0	-1
1	0	-1
1	0	-1

3×3

$=$

0	30	30	0
0	30	30	0
0	30	30	0
0	30	30	0

① **Parameter sharing:** A feature detector (such as a vertical edge detector) that's useful in one part of the image is probably useful in another part of the image.

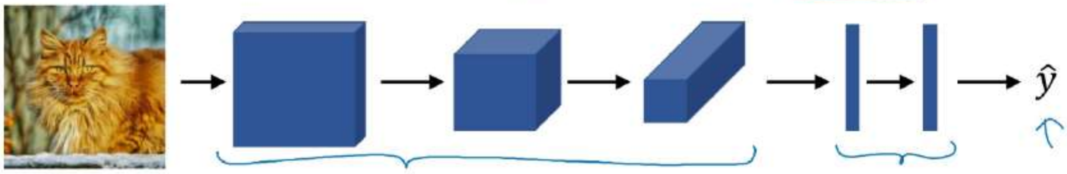
② **Sparsity of connections:** In each layer, each output value depends only on a small number of inputs.

Putting it together

이동 불변성을 포착하는 데에도 용이하다

Training set $(x^{(1)}, y^{(1)}) \ldots (x^{(m)}, y^{(m)})$.

$\boxed{w, b}$



Cost $J = \frac{1}{m} \sum_{i=1}^{m} \mathcal{L}(\hat{y}^{(i)}, y^{(i)})$

Use gradient descent to optimize parameters to reduce J