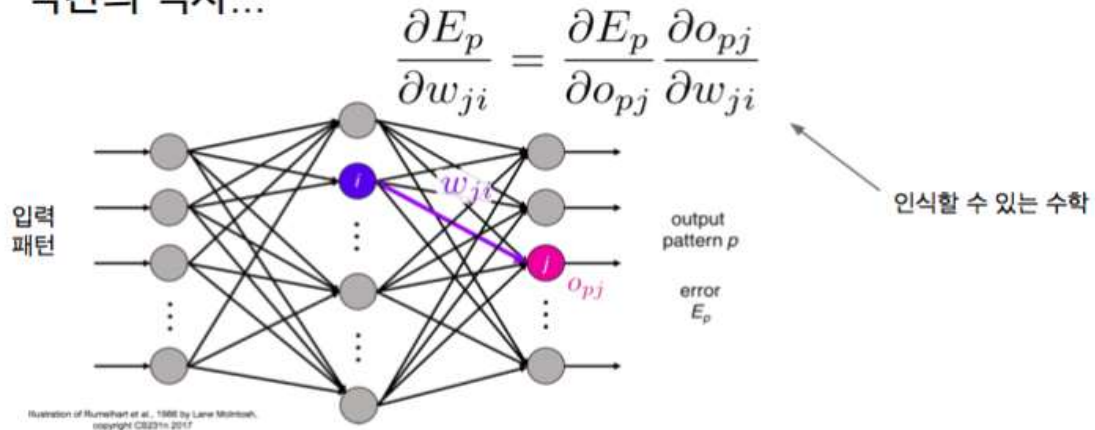


CS231n(5강)

최민우

1.A Bit History

약간의 역사...



Rumelhart et al., 1986: First time back-propagation became popular

- 발전 과정: 퍼셉트론 -> 아다린 (Adaline)과 마다린 (Madaline) -> 역전파 -> 심층 신경망

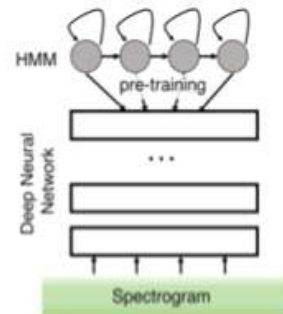
첫번째 강력한 결과

Acoustic Modeling using Deep Belief Networks

Abdel-rahman Mohamed, George Dahl, Geoffrey Hinton, 2010

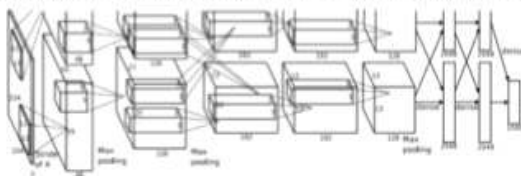
Context-Dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition

George Dahl, Dong Yu, Li Deng, Alex Acero, 2012



Imagenet classification with deep convolutional neural networks

Alex Krizhevsky, Ilya Sutskever, Geoffrey E Hinton, 2012

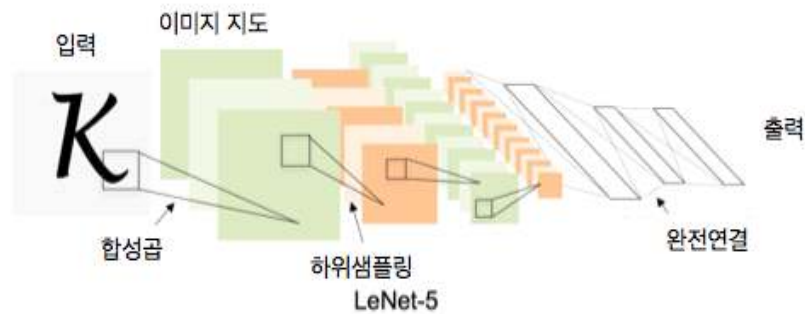


- 음성 인식: 제프리 힌튼 연구실의 어쿠스틱(acoustic) 모델링과 음성 인식에 대한 연구 -> 가장 강력한 결과
- 이미지 인식: 제프리 힌튼 연구실의 알렉스 키제프스키의 랜드마크(landmark) 논문 -> 이미지넷(ImageNet) 분류에 있어서, 매우 강력한 결과
- 합성곱 신경망을 활용한 콘브넷(ConvNet)이 모든 종류의 응용에서 광범위하게 받아들여짐

약간의 역사...

Gradient-based learning applied to document recognition

[LeCun, Bottou, Bengio, Haffner 1998]



- 1998년, 얀 르쿤이 역전파와 경사기반 학습을 적용하여, 합성곱 신경망을 훈련시키는 첫 번째 예를 보여줌
- 문서 인식을 잘 수행함
- 우편번호와 숫자인식을 매우 잘하여, 우체국에서 우편번호 인식에 널리 사용됨

약간의 역사...

ImageNet Classification with Deep Convolutional Neural Networks

[Krizhevsky, Sutskever, Hinton, 2012]

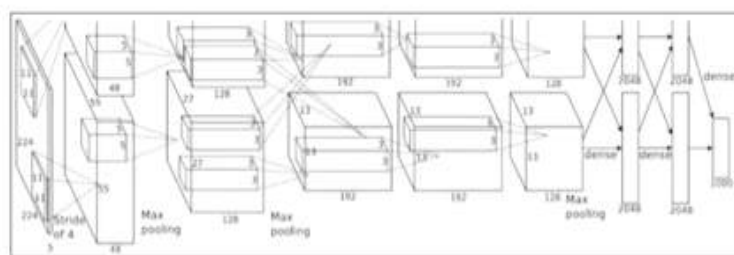
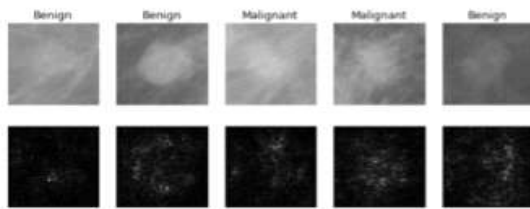


Figure copyright Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton, 2012. Reproduced with permission.

“AlexNet”

- 2012년, 알렉스 크리제프스키가 합성곱 신경망을 현대적인 형태로 제안 -> 알렉스넷(AlexNet)
- 얀 르쿤의 합성곱 신경망보다 더 크고 더 깊게 확장
- GPU의 병렬 컴퓨팅 파워를 이용할 수 있게 되었음
- 이미지넷, 웹 이미지 등에서 가용한 많은 데이터를 이용할 수 있게 되었음

오늘날로 빨리 감기: 콘브넷은 모든 곳에 있음



[Levy et al. 2016]

Figure copyright Levy et al. 2016.
Reproduced with permission.



[Dieleman et al. 2014]

From left to right: public domain, by NASA, usage permitted by ESA/Hubble, public domain by NASA, and public domain.



[Sermanet et al. 2011]
[Ciresan et al.]

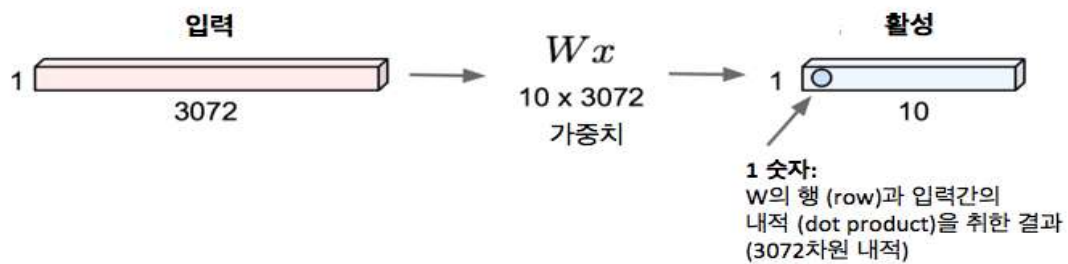
Photos by Lane McIntosh.
Copyright CS223n 2017.

- 콘브넷(ConvNet)을 이용하여, 의료 이미지의 해석과 진단, 은하 분류, 거리 표지판 인식 등

2. 콘브넷(ConvNet)

완전 연결 계층 (Fully Connected Layer)

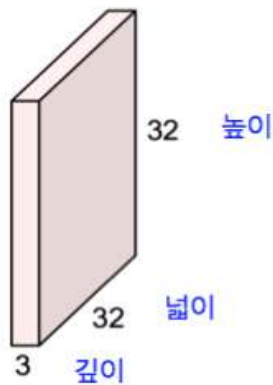
32 X 32 X 3 이미지 → 3072 X 1로 늘림



- 완전연결층(Fully Connected Layer)
- 32 * 32 * 3 이미지를 3072 * 1로 늘림(3차원 -> 1차원)
- 10개 행과 3072차원 입력을 내적(Dot Product)

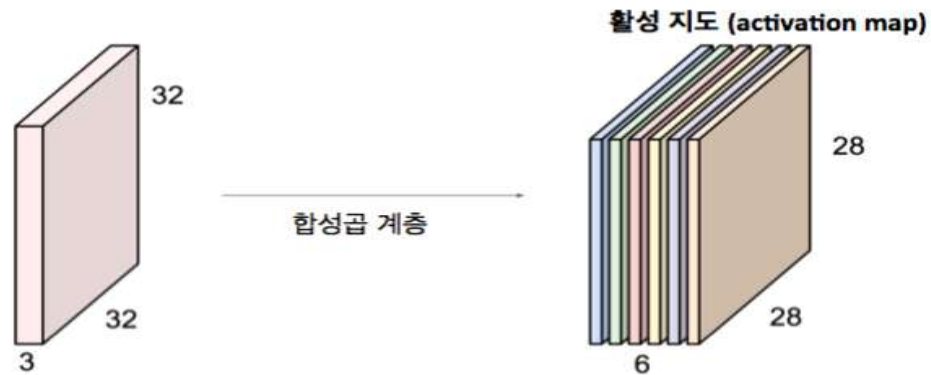
합성곱 계층 (Convolution Layer)

32x32x3 이미지 → 공간 구조 유지



- 합성곱 계층(Convolutional Layer)과 완전연결층의 중요한 차이점: 공간구조 유지
- 32 x 32 x 3 이미지를 유지: 3차원 구조로 입력

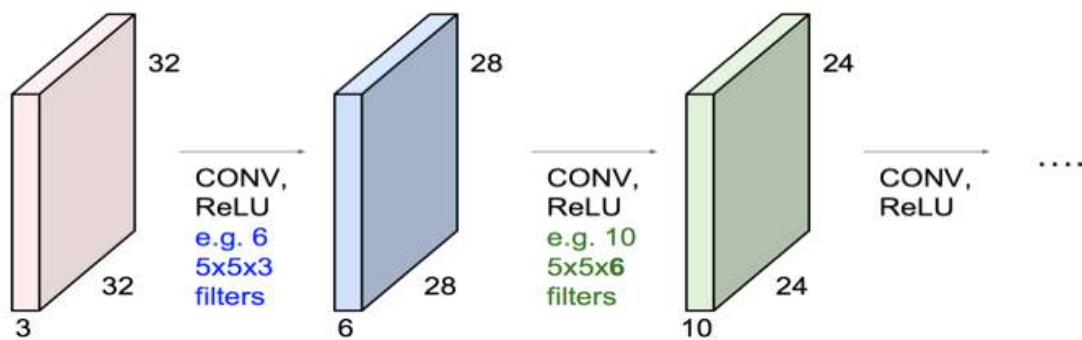
예를 들어, 6개의 5x5 필터가 있다면, 6개의 별도의 활성 지도를 갖게 됨:



우리는 이것을 쌓아서 28x28x6의 “새로운 이미지”를 얻음!

- 필터(Filter)
- $32 * 32 * 3 \rightarrow 28 * 28 * 6$

미리보기: 콘브넷 (ConvNet)은 여러 합성곱 계층들 (convolution layer)의 결과이고, 활성 함수 (activation function)들이 중간에 끼어 있음

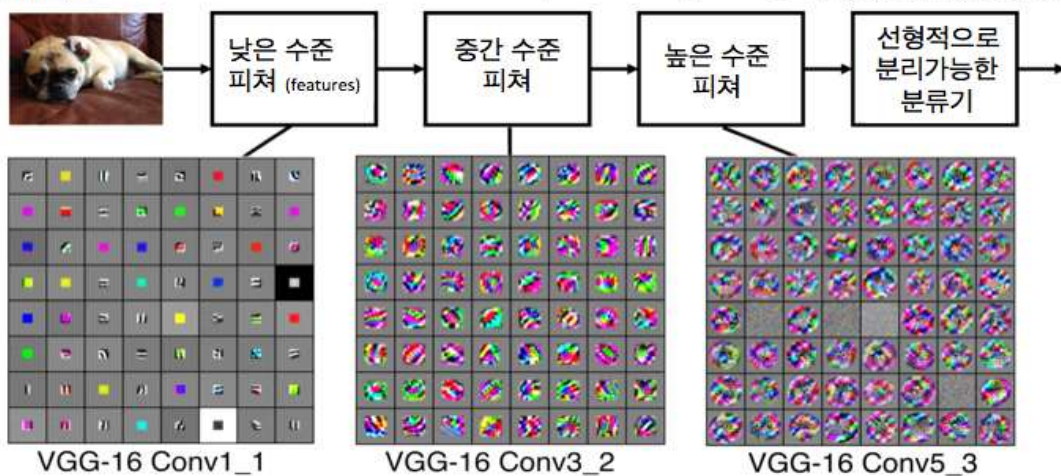


- 활성화 함수들과 함께 배치(ReLU)

미리보기

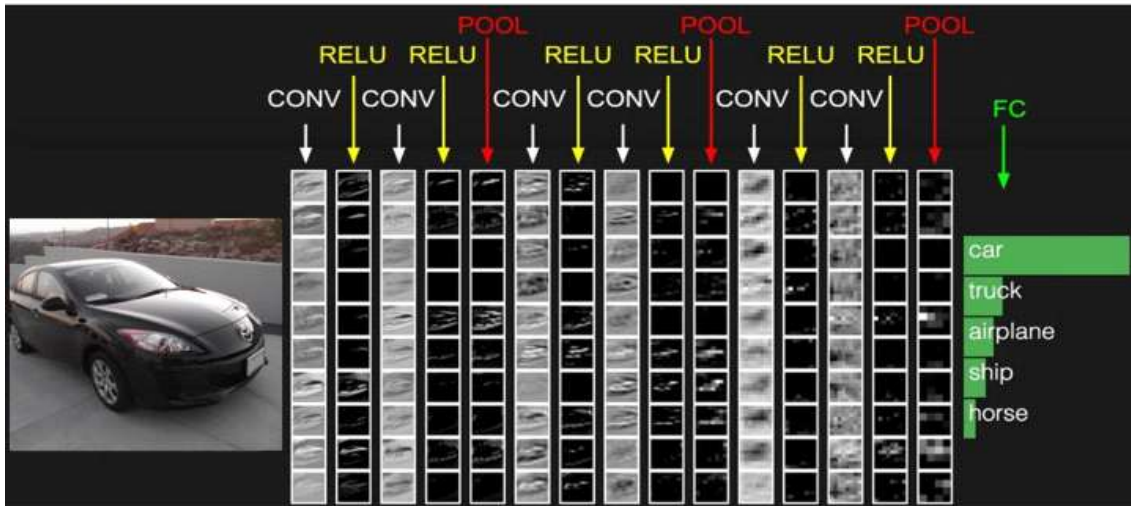
[Zeiler and Fergus 2013]

Visualization of VGG-16 by Lane McIntosh. VGG-16 architecture from [Simonyan and Zisserman 2014].



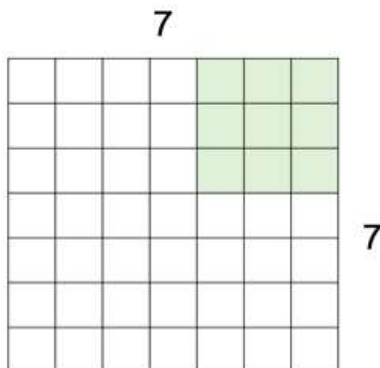
- 낮은 수준 피쳐(엣지) -> 중간 수준 피쳐(모서리, 방울) -> 높은 수준 피쳐(복잡한 개념)

미리보기:



- 합성곱 신경망의 전체적인 구조(CONV, RELU, POOL)

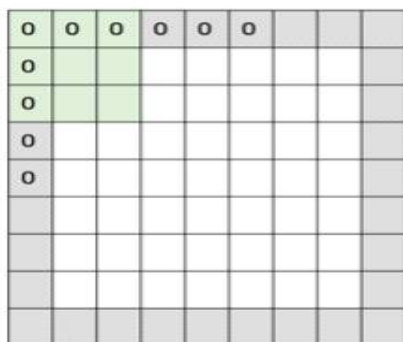
공간 차원 (spatial dimension) 더 자세히 보기:



7x7 입력 (공간적으로)
3x3 필터 가정
스트라이드 (stride) 2 적용
=> 3x3 출력

- 스트라이드(Stride) : 몇 칸을 이동할 것인가? 2(2칸씩 이동)

실제예선: 흔히 경계를 영으로 패딩 (zero pad)

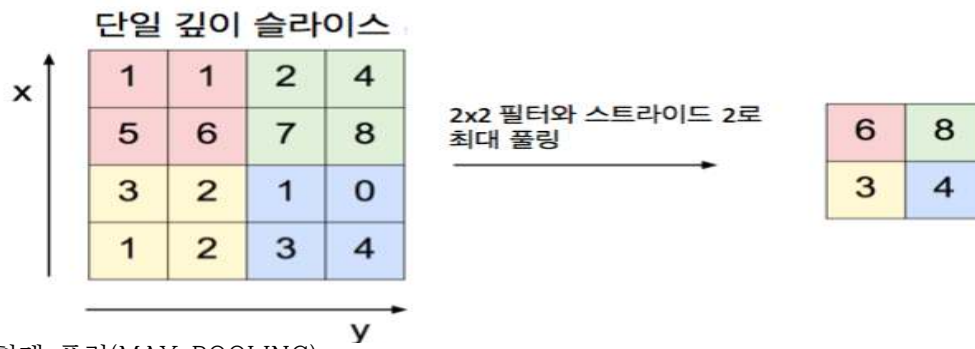


예. 7x7 입력
3x3 필터에 스트라이드 1을 적용
1 픽셀로 경계를 패딩 => 출력은?

7x7 출력!

- 제로패딩(Zero Padding): 원하는 크기를 얻기 위하여, 0으로 주변을 채우는 것(7 * 7)

최대 풀링 (MAX POOLING)



- 최대 풀링(MAX POOLING)