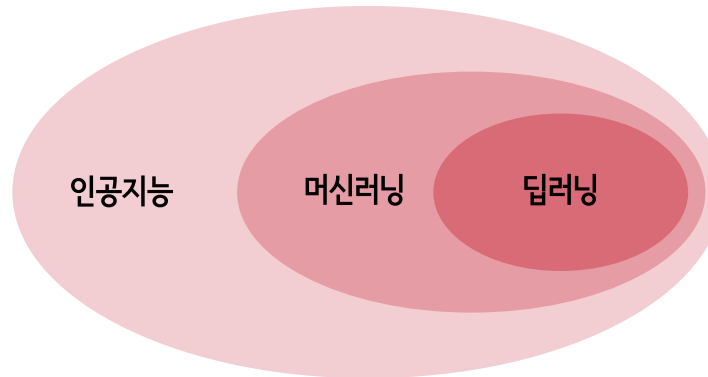


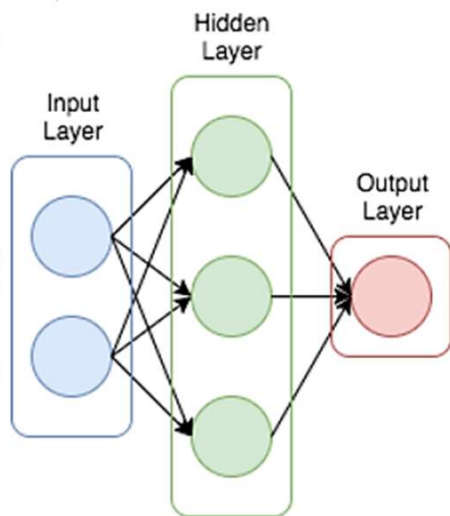
딥러닝

- 딥러닝(Deep Learning)은 여러 층으로 구성된 신경망을 사용하여 머신러닝 학습을 수행하는 심층 학습
 - 머신러닝에서는 학습하려는 데이터의 여러 특징 중에서 어떤 특징을 추출할지를 사람이 직접 분석하고 판단
 - 딥러닝은 **기계가 자동으로 대규모 데이터에서 중요한 패턴 및 규칙을 학습**하고, 이를 토대로 의사결정이나 예측 등을 수행
 - 딥러닝 성과
 - 사람과 비슷한 수준의 이미지 분류 및 음성 인식, 필기 인식
 - 향상된 기계 번역과 자연스러운 음성 합성(Text-To-Speech), 똑똑해 진 웹 검색 엔진의 결과
 - 자연어 질문에 대답하는 능력
 - 사람과 비슷한 수준의 자율주행 능력
 - 사람을 능가하는 바둑 실력

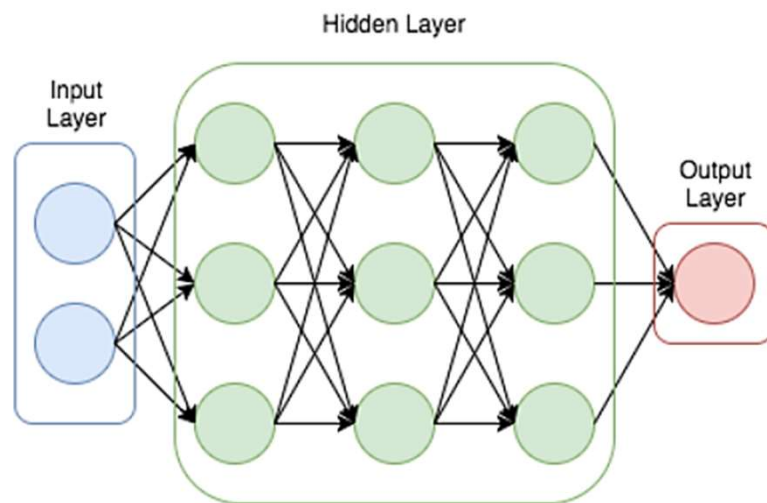


심층 신경망

- 심층 신경망(DNN: Deep Neural Network)은 더욱 더 복잡한 모델링을 위해 여러 개의 은닉층을 가지고 있는 신경망
 - 입력층과 출력층 사이에 여러 개의 은닉층들로 이뤄진 신경망
 - 일반적인 신경망과 같이 복잡한 비선형 관계(non-linear relationship) 모델링



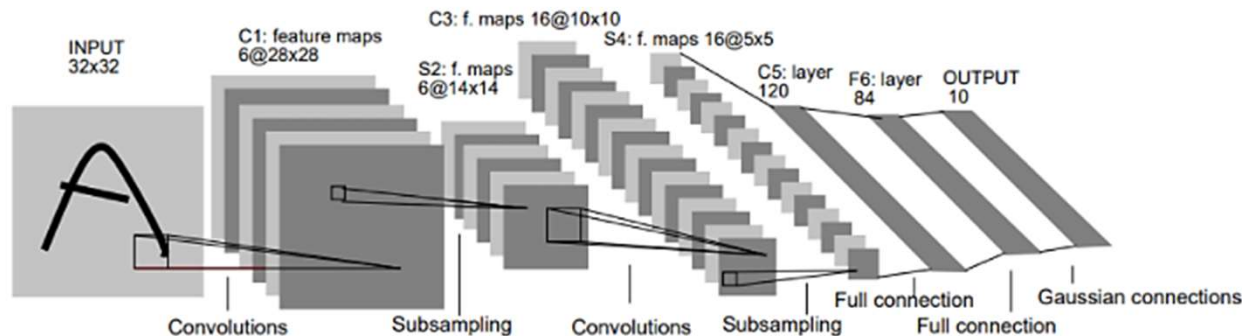
신경망



심층 신경망

합성곱 신경망

- 합성곱 신경망(CNN Convolutional Neural Network)은 컬러 이미지 같은 다차원 배열 처리에 특화
 - 일반적인 신경망은 이미지 데이터를 그대로 처리하는 반면 합성곱 신경망은 **이미지에서 특징을 추출해 처리**
 - 컨볼루션 연산을 통해 이미지의 공간적인 정보를 유지하고, 일반적인 신경망(Fully connected Neural Network) 대비 연산량을 획기적으로 줄임
 - 이미지 분류 알고리즘을 개발할 때 필터링 기법을 이용하면 분류의 정확도를 향상시킬 수 있음
 - 기존 이미지 처리는 사람의 직관이나 반복적인 실험을 통해 알고리즘에 이용될 필터 결정
 - **합성곱 신경망은 행렬로 표현된 필터의 각 요소가 데이터 처리에 적합하도록 자동으로 학습** ➔ 필터가 곧 신경망의 변화량 매개변수

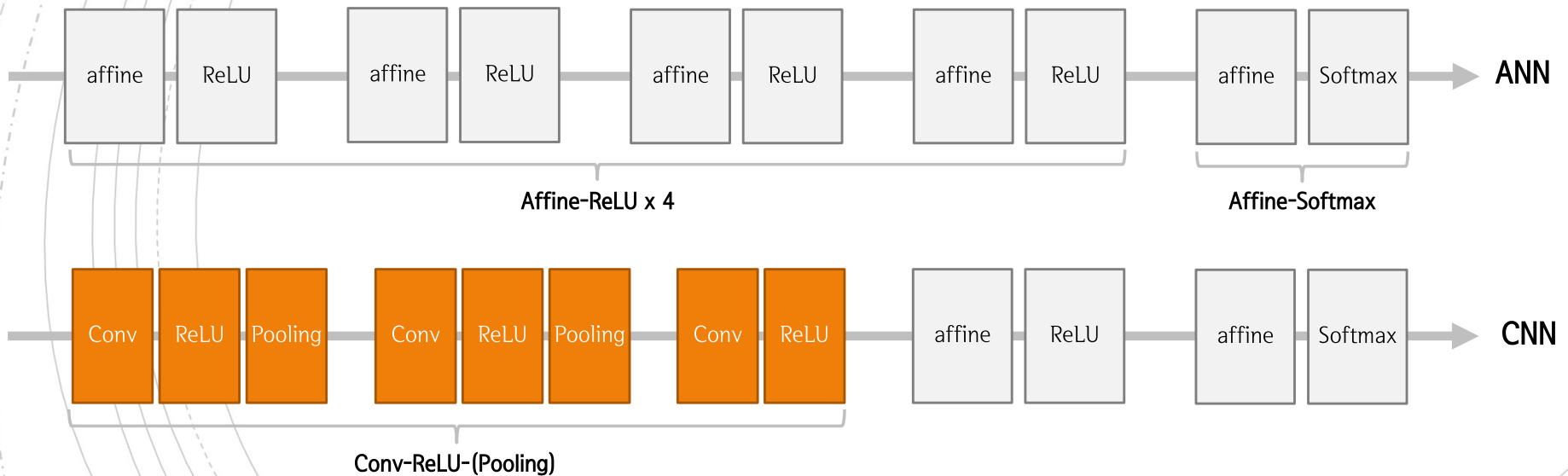


초기 합성곱 신경망 (LeNet-5)



합성곱 신경망

- 합성곱 신경망 구조
 - 일반적인 신경망은 서로 인접한 레이어의 모든 노드가 연결된, 완전 연결(fully-connected) 상태
 - 어파인으로 명시된 완전 연결 연산과 렐루 함수와 같은 비선형 활성화 함수의 조합으로 신경망 구성
 - 합성곱 신경망은 합성곱 레이어와 생략가능한 풀링(pooling) 레이어를 완전 연결 레이어 앞에 추가
 - 원본 이미지에 필터링 기법(컨볼루션 연산)을 적용한 후 이미지 분류 연산 수행
 - 풀링 레이어는 각 이미지의 국소적인 부분을 하나의 대표적인 스칼라 값으로 변환하여 이미지 크기를 줄임
 - '합성곱-풀링' 다음(출력과 가까운 곳)은 '어파인-렐루', 마지막은 '어파인-소프트맥스' 조합을 그대로 사용

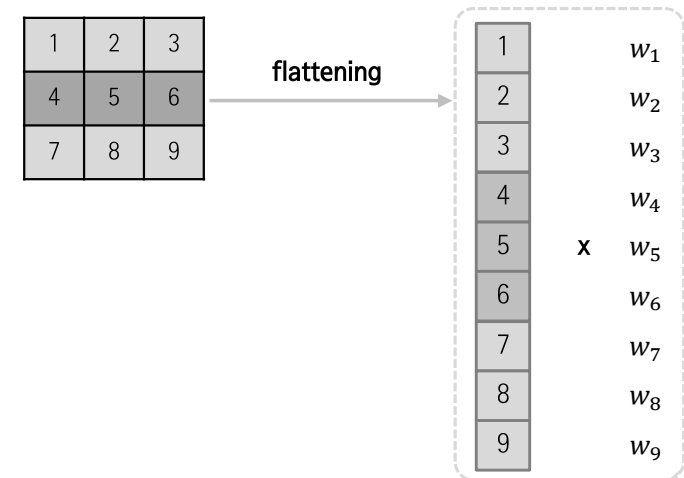
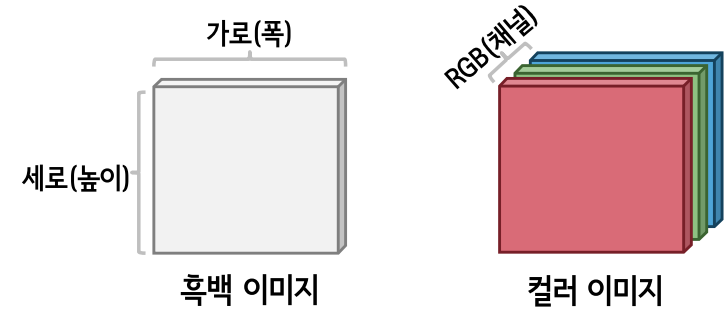
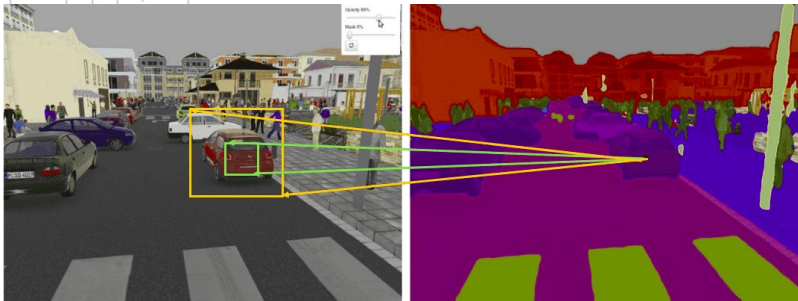




합성곱 신경망

■ 컨볼루션 레이어의 필요성

- 이미지는 픽셀 단위로 가로, 세로 크기와 채널로 구성
 - 흑백 이미지 채널은 1, 컬러 이미지는 3 (R, G, B 별로 자신의 강도 표현)
 - 흑백 이미지의 정형 데이터화: pixel x pixel x 1
 - 컬러 이미지의 정형 데이터화: pixel x pixel x 3
 - 심층 신경망으로 'c x h x w' 이미지 분류
 - 'c x h x w' 배열을 펼쳐서(Flattening) 각 픽셀을 곱해 은닉 레이어로 결과 전달
 - 이미지는 픽셀 간 밀접한 상관 관계가 있으나, 플래트닝하면 공간적 구조가 사라짐
 - 이미지 데이터의 공간적 구조(면적)를 유지하기 위해 컨볼루션 레이어 필요
- 수용영역(Receptive field)은 특성을 생성하는 입력 영역의 크기
 - 입력 영역에 대한 출력 기능(모든 레이어) 연관
 - 작으면 물체 인식이 떨어지고, 크면 연산 시간 증가

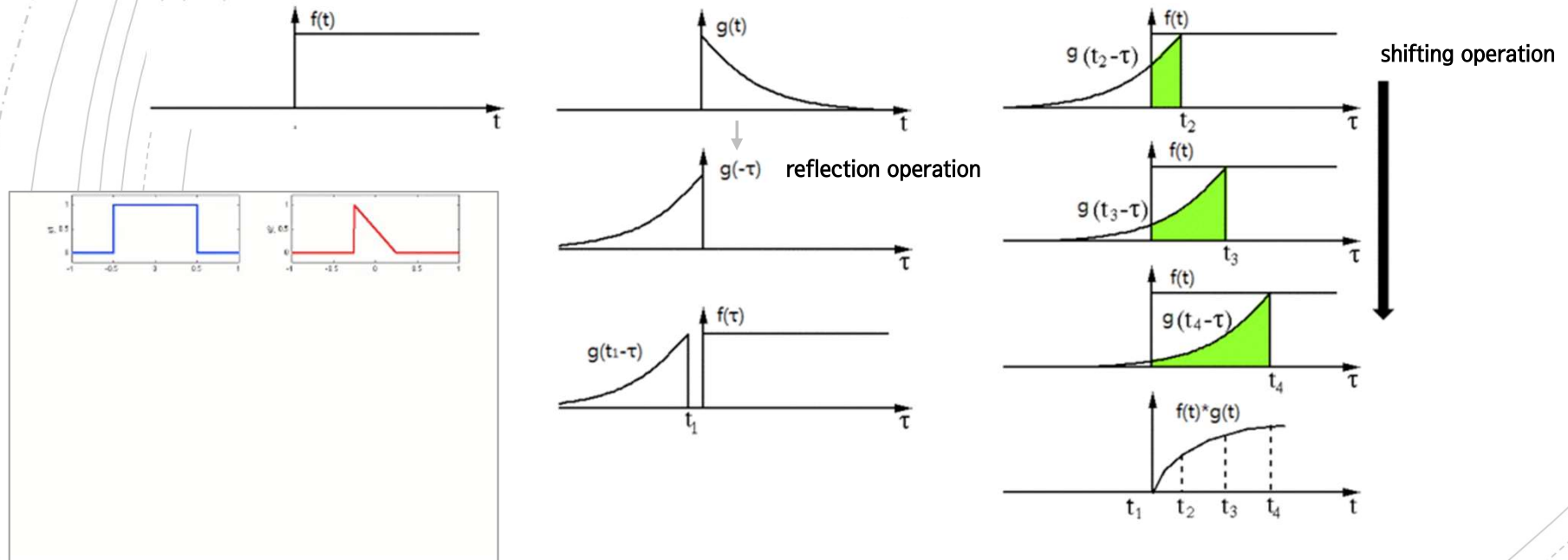




합성곱 신경망

■ 컨볼루션 연산

- 컨볼루션 연산은 두 함수 f , g 중 하나를 뒤집은(reflection) 후 하나씩 밀면서(shift) 겹치는(overlap) 면적을 계산(적분)해 나가는 것
 - $y(t) = (f * g)(t) = f(t) * g(t) = \int f(\tau)g(t - \tau)d\tau = \int g(t)f(t - \tau)d\tau = g(t) * f(t) = (g * f)(t)$
 - $f(\tau)g(t - \tau)$ 와 $f(t - \tau)g(t)$ 는 결과가 같으며, 뒤집기 연산을 제외하면 합성곱 신경망에서 사용하는 교차 상관관계(cross-correlation)





합성곱 신경망

- 컨볼루션 연산 사례

- $f = 4t^3 + 3t^2 + 6 \rightarrow [4, 3, 0, 6]$
- $g = 5t^2 + 3t + 1 \rightarrow [5, -3, 1]$

$$\begin{array}{r} 4 \ 3 \ 0 \ 6 \\ \times 5 \ -3 \ 1 \\ \hline \end{array}$$

$$20 \ 15 \ 0 \ 30$$

$$\begin{array}{r} 4 \ 3 \ 0 \ 6 \\ \times 5 \ -3 \ 1 \\ \hline 20 \ -9 \ 0 \end{array}$$

$$\begin{array}{r} 4 \ 3 \ 0 \ 6 \\ \times 5 \ -3 \ 1 \\ \hline \end{array}$$

$$20 \ 15 \ 0 \ 30 \\ -12 \ -9 \ 0 \ -18$$

$$\begin{array}{r} 4 \ 3 \ 0 \ 6 \\ \times 5 \ -3 \ 1 \\ \hline 20 \ -9 \ 0 \\ 15 \ 0 \ 6 \end{array}$$

$$\begin{array}{r} 4 \ 3 \ 0 \ 6 \\ \times 5 \ -3 \ 1 \\ \hline \end{array}$$

$$20 \ 15 \ 0 \ 30 \\ -12 \ -9 \ 0 \ -18 \\ 4 \ 3 \ 0 \ 6$$

$$\begin{array}{r} 4 \ 3 \ 0 \ 6 \\ 5 \ -3 \ 1 \\ \hline 20 \ -9 \ 0 \\ 15 \ 0 \ 6 \\ \hline 11 \ 21 \end{array}$$

$$\begin{array}{r} 4 \ 3 \ 0 \ 6 \\ \times 5 \ -3 \ 1 \\ \hline \end{array}$$

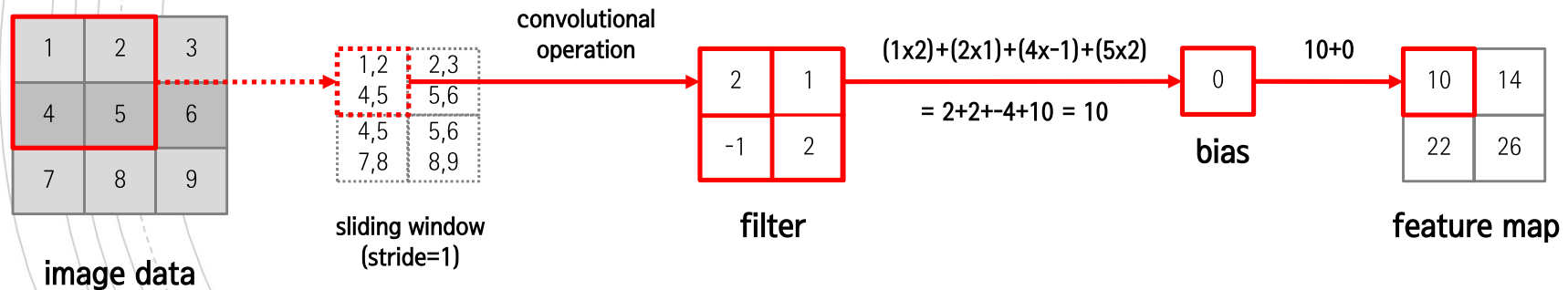
$$\begin{array}{r} 20 \ 15 \ 0 \ 30 \\ -12 \ -9 \ 0 \ -18 \\ + \quad 4 \ 3 \ 0 \ 6 \\ \hline 20 \ 3 \ -5 \ 33 \ -18 \ 6 \end{array}$$



합성곱 신경망

■ 컨볼루션 레이어

- 컨볼루션 레이어에서 자동 계산되는 변화율 매개변수는 수용영역에 연관되는 작은 행렬로 필터(filter), 커널(kernel), 윈도우(window)로 불림
 - 슬라이딩 윈도우(sliding window) 방식으로 입력값과 필터의 컨볼루션 연산을 수행하면 특성 맵이 만들어지고, 이를 활성화 함수를 통해 은닉층으로 전달
 - 이미지 데이터는 서로 인접한 픽셀 사이 유사성이 높으므로, 픽셀 전체를 다루는 것보다 지역 단위 영역에 대한 특성을 다루는 것이 효과적
 - 슬라이딩 윈도우는 고정된 크기의 윈도우가 이동하면서 윈도우 내에 있는 데이터를 이용해 문제 해결
 - 스트라이드(stride)는 윈도우 이동 간격
 - 필터와 슬라이딩 윈도우를 이용한 컨볼루션 연산으로 연산 속도 증가
 - 전체 데이터 중 필터에 대응하는 일부 데이터와 단순 행렬의 요소 곱셈 및 전체 덧셈을 수행하므로 연산 속도가 빠름
 - 전체 픽셀 수 만큼의 변화율 매개변수 대신 작은 필터가 변화율 매개변수이므로, 매개변수의 갯수가 적어 연산 속도가 빠름
 - 필요에 따라 편향(bias)을 추가하면 특성의 요소에 편향이 더해 짐
 - 특성을 신경망으로 학습하면서 필터 최적화





합성곱 신경망

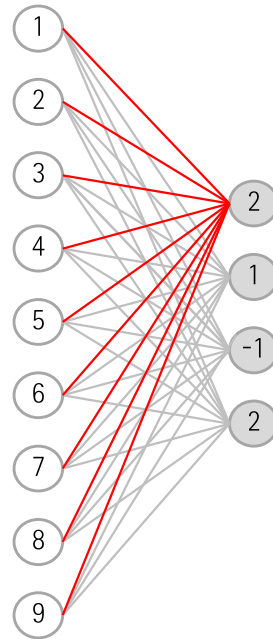
- 컨볼루션 과정은 데이터의 일부만 은닉층으로 연결됨(local connectivity)
 - 어파인은 하나의 가중치가 모든 노드에 적용되지만 컨볼루션은 하나의 가중치가 일부 노드에만 적용됨

1	2	3
4	5	6
7	8	9

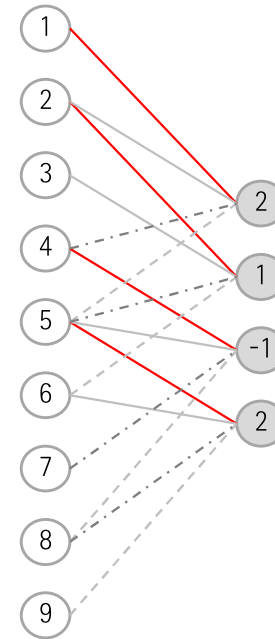
image data

2	1
-1	2

filter



fully connected layer



convolutional layer

합성곱 신경망

-

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

2	1
-1	2

4	4	4
4	4	4
4	4	4

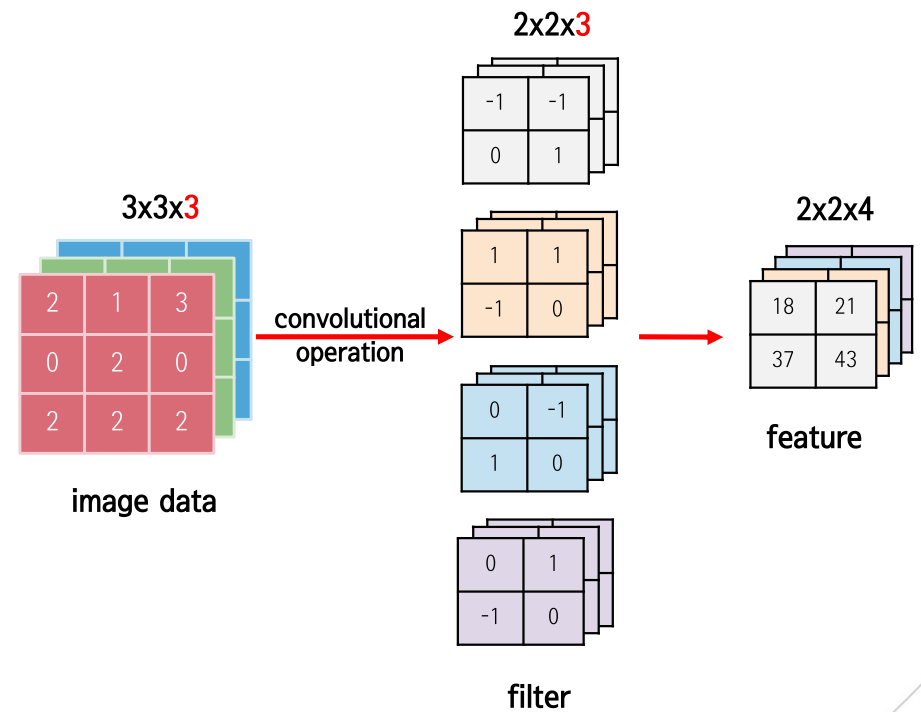
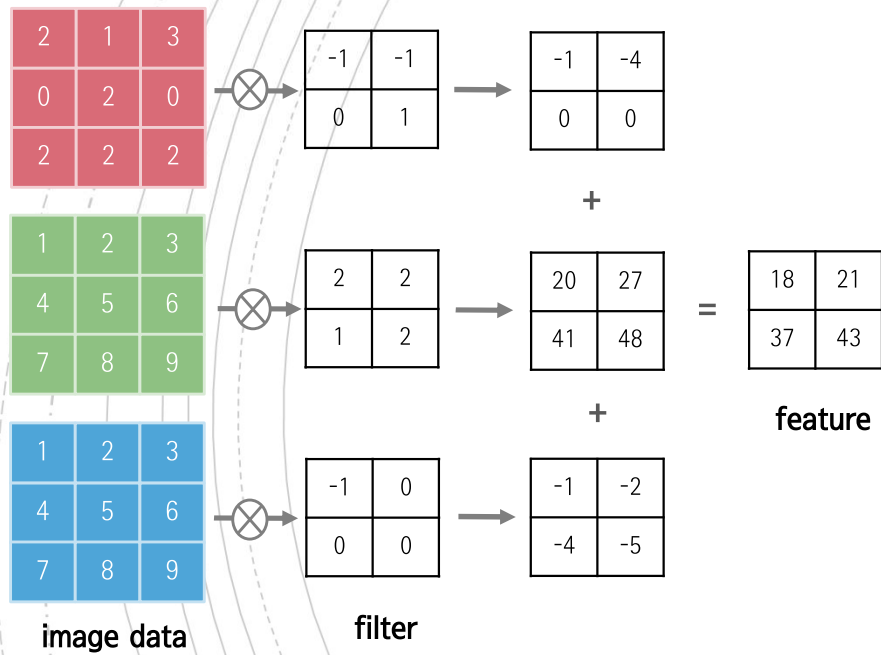
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	1	2	3	0	0
0	0	4	5	6	0	0
0	0	7	8	9	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

padding size=2



합성곱 신경망

- 컬러 이미지는 채널이 3개이므로 RGB 각각 다른 가중치로 컨볼루션 연산 수행
 - 필터가 2개 이상이면 특성 맵도 필터 개수만큼 만들어짐
 - 이미지 채널과 필터 채널 별로 컨볼루션 연산을 수행한 후 채널을 모두 더해 특성 맵 구성

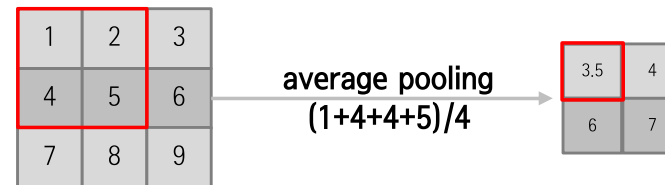




합성곱 신경망

풀링 및 플래터링 레이어

- 풀링(Pooling) 레이어는 데이터의 공간적인 특성을 유지하면서 크기를 감소시키기 위해 연속적인 컨볼루션 레이어 사이에 삽입
 - 서브샘플링(sub-sampling)으로도 불리며(일반적으로 윈도우 크기와 스트라이드 값을 같게 설정), 학습할 매개변수가 없음
 - 역할
 - 특정 위치에서 주도적 역할을 하는 특징 추출하거나 전체를 설명하는 특징 추출
 - 학습할 가중치를 크게 줄여, 과적합 문제(overfitting) 해결에 도움을 줌
 - 값 추출 방식에 따른 구분
 - 최대 풀링(Max Pooling): 윈도우 영역에서 가장 큰 값을 추출하며, 이동/회전 변환에 따른 출력값의 영향을 덜 받음
 - 평균 풀링(Average Pooling): 윈도우 영역의 평균 추출



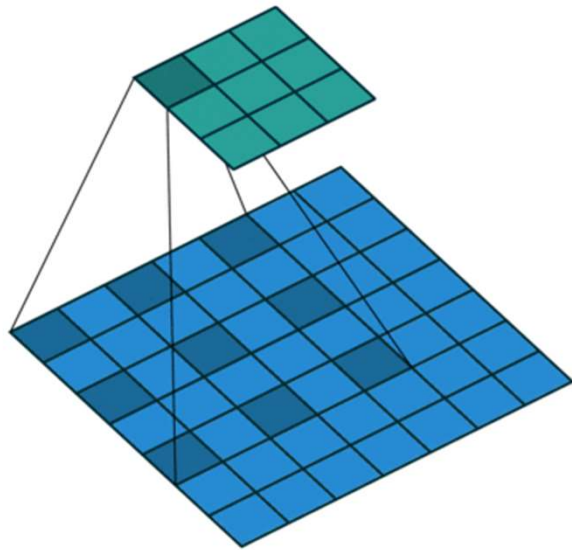
- 플래터닝 레이어는 '컨볼루션-풀링' 레이어를 통해 추출한 특성을 '어파인-렐루' 레이어가 처리할 수 있도록 펼쳐서 연결
 - 플래터닝 레이어 다음은 일반적인 신경망 모델과 같음



합성곱 신경망

- 커널 요소 팽창(dilated)과 컨볼루션 연산
 - 수용영역 확장 없이 연산량을 줄이고, 전체적인 특징을 잘 잡아낼 수 있어 객체의 형태를 구분하는 세그먼테이션(Segmentation)에 유리

`nn.Conv2d(3, 6, 3, dilation=2)`



출처: A guide to convolutional arithmetic (Dumoulin et al. 2016)

