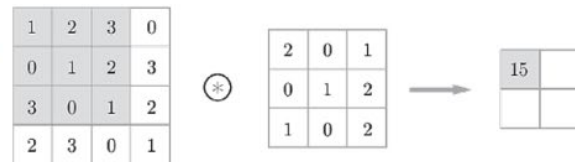
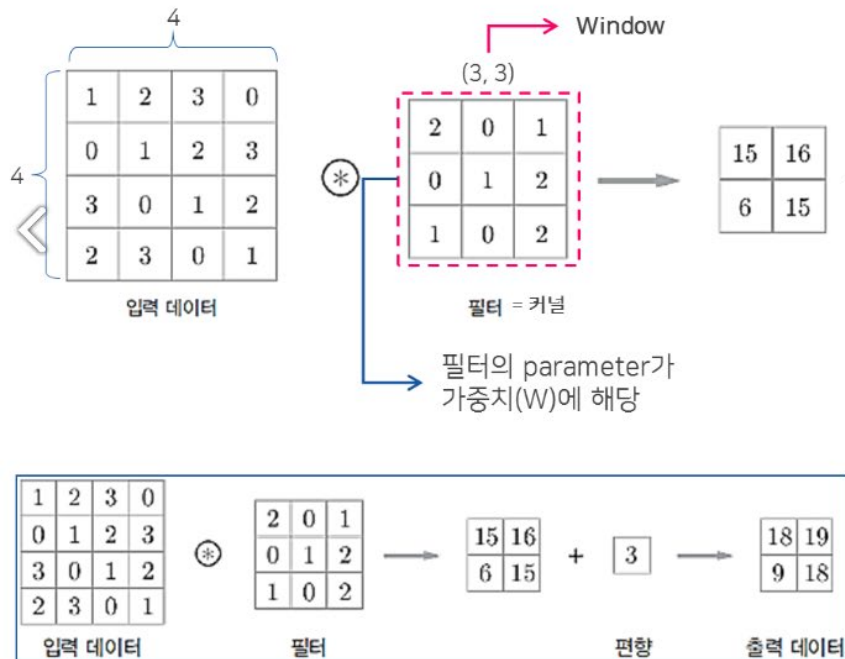


HW#2 Convolution 시각화

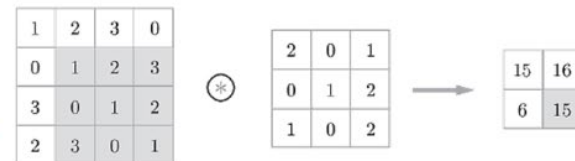
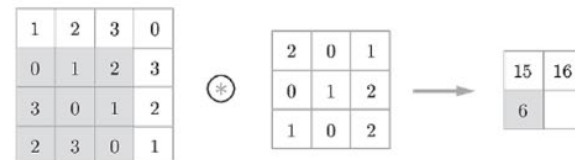
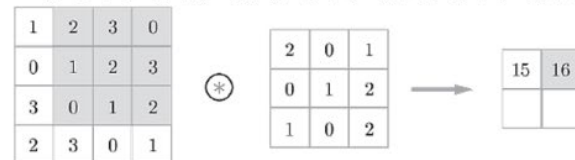
- 첫 페이지에는 학번, 이름, HW2 버튼이 존재한다.
 - HW2 버튼을 누르면 다음 페이지로 이동한다.
- 두번째 페이지에서는 아래와 같이 Convolution 연산을 시각화한다.
 - 2차원 형태의 숫자 csv 파일을 받는다. (예시 코드로 구현됨, conv_prob.html)
- 커널에 대한 입력을 csv 형식으로 textarea를 통해 입력할 수 있으며, 3x3 으로 제한된다.
- Apply Convolution 버튼을 누르면 convolution 연산의 output 출력 과정을 line 당 5ms 마다 재생되도록 한다.
 - 기존 이미지 위에 덮어쓰기 방식으로 구현하여 이미지가 점차 위에서부터 바뀌는 형태로 구현한다.
 - convolution 연산은 for 를 활용하여도 됨
- 해당 과제는 React 를 사용하여 구현한다.
- 제출은 github 로 수행한다 (<https://classroom.github.com/a/LF6iXLn1>)
 - 채점 시 감안하여야할 사항은 첫 페이지에 작성한다.
- 채점은 clone 후, npm install 명령어 이후 npm start 로 수행함
 - 5점 : 동작이 성공적이며, 주석이 잘 남겨져있음
 - 3점 : 동작은 성공적이나, 주석이 없고, 권장하지 않는 문법 사용. 가끔 성공적으로 동작함
 - 1점 : 동작에 실패함
 - 0점 : 미제출

Convolution 연산

- 합성곱

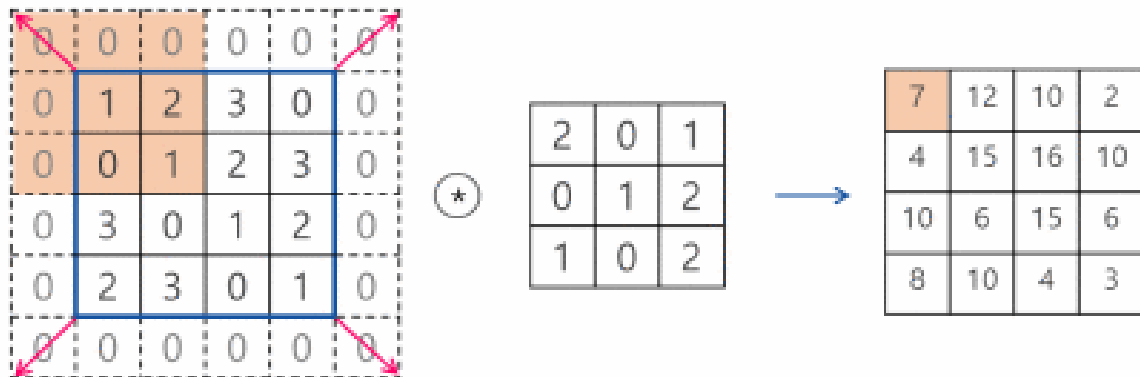


- 단일 곱셈-누산(FMA, Fused Multiply-Add)
 $1 * 2 + 2 * 0 + 3 * 1 + 0 * 0 + 1 * 1 + 2 * 2 + 3 * 1 + 0 * 0 + 1 * 2 = 15$



Convolution 연산

- 합성곱
 - 예시는 10번째 요소까지만 보여줌



Convolution 연산

- 결과

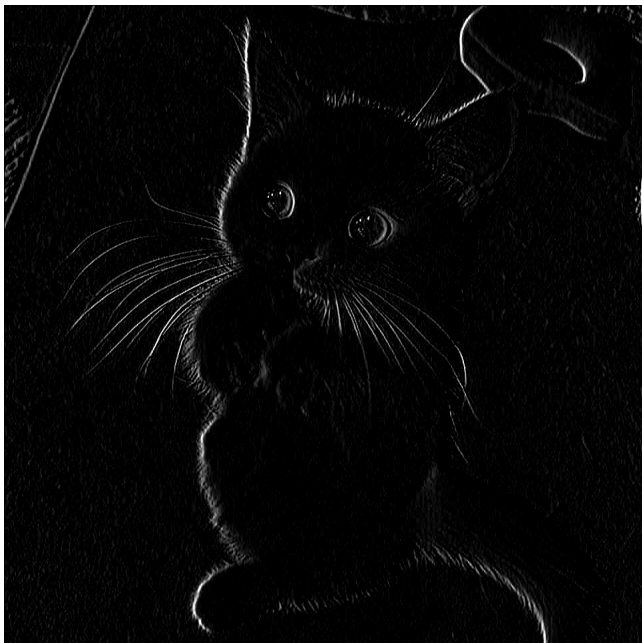


Convolution 연산

- 결과

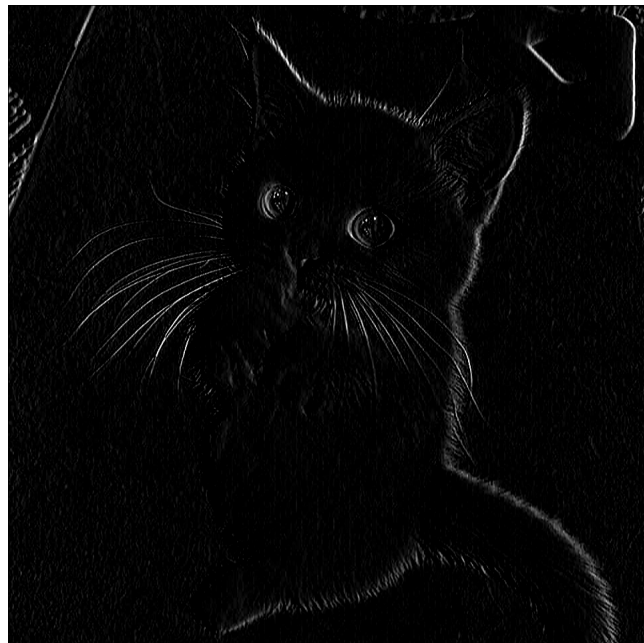
-1, 0, 1
-1, 0, 1
-1, 0, 1

Apply Convolution



1, 0, -1
1, 0, -1
1, 0, -1

Apply Convolution

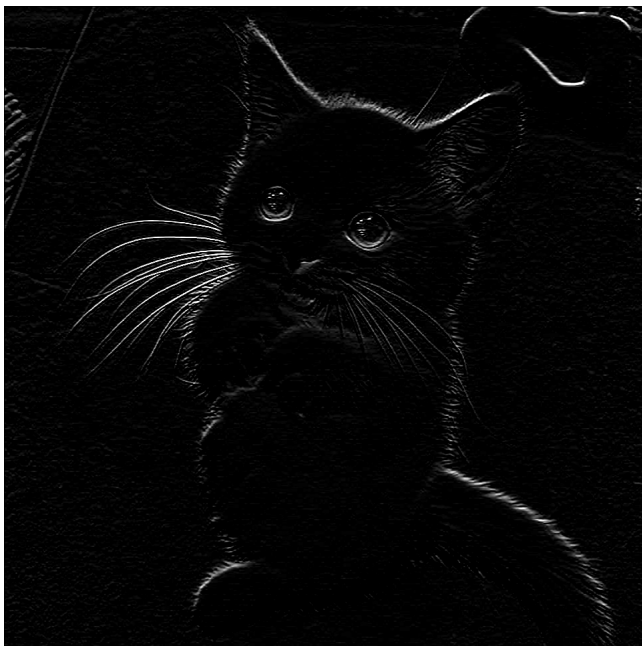


Convolution 연산

- 결과

-1, -1, -1
0, 0, 0
1, 1, 1

Apply Convolution



1, 1, 1
0, 0, 0
-1, -1, -1

Apply Convolution

