

## (1) 번 숙제

conv\_2d\_fileter: 4중 for문을 사용해 작성했다. 첫 2개 for문은 original\_im의 row와 column에 대해, 그 다음 2개의 for문은 filter의 row와 column에 대해 진행하게 작성했다.

conv\_1d\_filter\_x: Example 을 활용하여 만들었다. original\_im 의 row 를 zero-pad 한 original\_im\_pad 를 만들고, filter 또한 적절히 활용하여(Example 과 같은 방법으로) H 를 만들고, 둘을 행렬곱 하여 result\_im 을 만들고 return 해주도록 하였다. 다만 기존 example 에서는 size(x)가 5x1 인데 반해 size(y)가 1x5 였다. 따라서 이를 크기를 일치히시키기 위해 return 하기 전 다시 '를 통해 result\_im 이 original\_im 과 같은 크기를 갖도록 하였다.

conv\_1d\_filter\_y: original\_im 와 filter 을  $\text{rot90}(\sim, 1)$  : (90)도 회전시킨 뒤, 회전시킨 값을 conv\_1d\_filter\_x 에 넣어 result\_im\_rot 을 구하였다. 그리고 이를 다시 (-90)도 회전시켜 원래 방향과 맞도록 하였다.

## (2)번 숙제

myFT: 기존 코드를 활용하였다.

cropCenter: "row\_start = floor(row\_freq/4)+1", "row\_end = floor(row\_freq/4\*3)"로 가운데 시작과 끝점을 지정해주었다. column 에 대해서도 같은 작업을 반복한 뒤, freq\_1 를 row\_freq 와 같은 크기를 갖는 0 행렬로 만들고, 위의 freq 의 row, column 의 start 와 end 사이 값만 freq\_1 에 넣어 return 하였다.

cropBoundary: cropCenter 과 유사하다. freq\_2 가 freq 의 값을 전체 가진 뒤, start 와 end 사이 값만 0 으로 바꾸어 return 하였다.

myIFT: "ifft2(ifftshift(ifftshift(freq,2),1))"로, inverse shift 를 먼저 해주고, inverse transform 을 하여 이미지를 구하고 return 하였다. myFT 의 역순이다.