

# 영상처리

## - DoG filter -

제출일자	2021.04.18
분 반	01
이 름	강인한
학 번	201701969

```
def get_DoG_filter(fsize, sigma):

    msizeDivTwo= fsize//2
    y,x = np.mgrid[-(msizeDivTwo):msizeDivTwo+1,-(msizeDivTwo):msizeDivTwo+1]

    DoG_x= (-x/(sigma*sigma)) * np.exp(-(x*x)+(y*y))/(2*sigma*sigma)
    DoG_y= (-y/(sigma*sigma)) * np.exp(-(x*x)+(y*y))/(2*sigma*sigma)

    return DoG_x, DoG_y
```

get\_DoG\_filter 함수는 미분된 가우시안 필터를 만드는 작업으로 fsize가 3이라면

x =

```
[[ -1  0  1]
 [ -1  0  1]
 [ -1  0  1]]
```

y =

```
[[ -1 -1 -1]
 [  0  0  0]
 [  1  1  1]]
```

이 된다.

그 후 미분한 가우시안 필터를 만들어 저장한다.

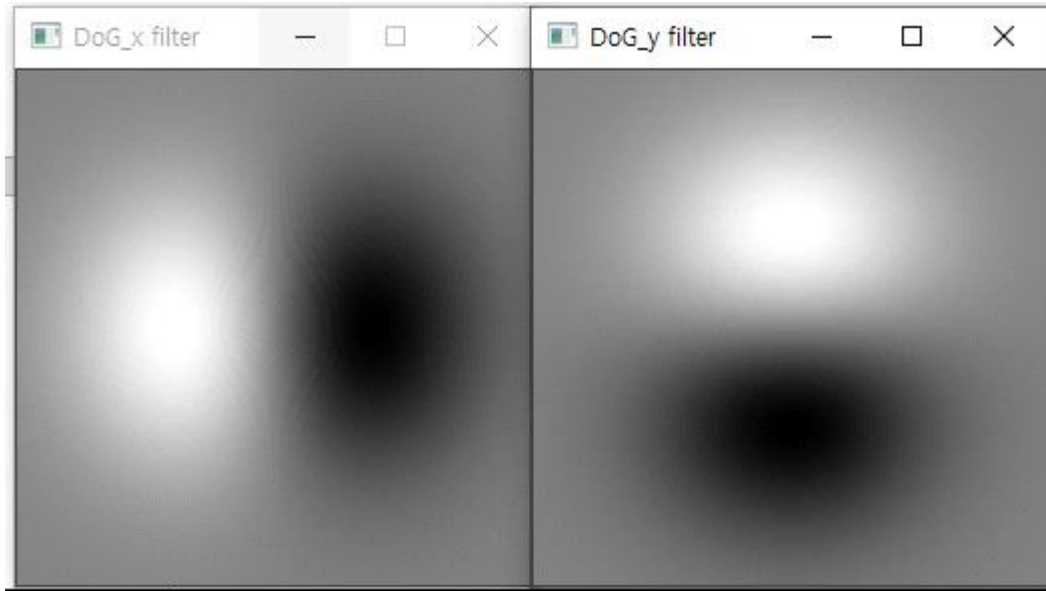
```
# DoG_x, DoG_y filter 확인
x, y = get_DoG_filter(fsize=256, sigma=50)

x= ((x-np.min(x))/ np.max(x-np.min(x))*255).astype(np.uint8)
y= ((y-np.min(y))/ np.max(y-np.min(y))*255).astype(np.uint8)
```

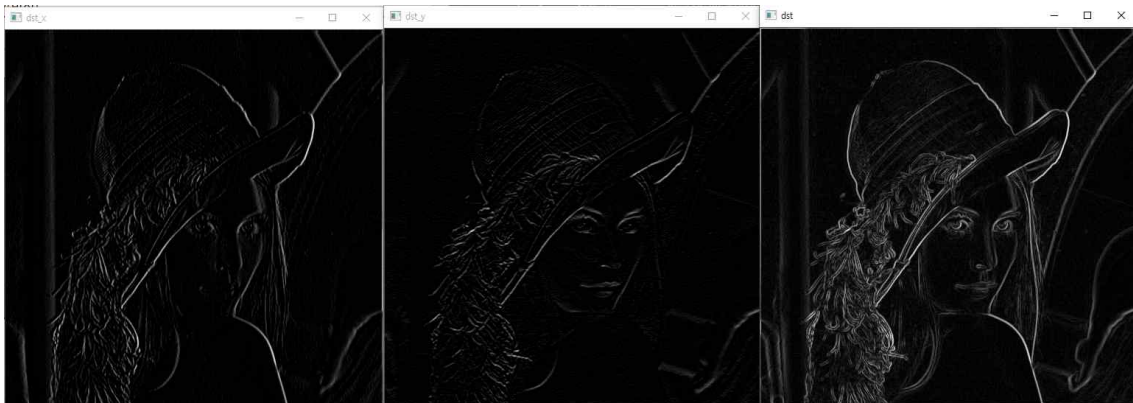
x와 y의 상대적인 밝기를 정해주는 과정이다. x에서 최소 x를 뺀 뒤 값의 차이가 가장 큰 값으로 나눠주고 255를 곱해주면 상대적인 밝기를 얻어낼 수 있다. (y도 같은 과정)

```
dst = np.sqrt(((dst_x * dst_x) + (dst_y * dst_y)))
```

magnitude를 계산하는 과정이다.



$\sigma = 50$



가우시안 필터를 이용해서 edge 부분을 찾는 게 재밌었습니다. 작성해야 할 코드는 어렵지 않아 과제를 이해하는데 시간이 더 오래 걸렸던 것 같습니다.