

문제2

2018316501 박민지

2. page 21. `imadjust`(입력영상, 입력범위_리스트, 출력범위_리스트, 감마) 구현

신규 조건

입력 영상은 `uint8`, `float`(0~1) 제한없어야 함,

입력영상은 1채널, 3채널 제한 없어야 함.

출력 영상은 입력 영상과 같은 데이터형이어야 함.

```
Path = 'data/'
```

```
Name = 'lenna.tif'
```

```
FullName = Path + Name
```

```
img = cv.imread(FullName)
```

--- `lenna.tif` 를 불러옵니다.

```
def imadjust(src, input_range=None, output_range=None, gamma=1):
```

```
    if input_range is None:
```

```
        input_range = (0,1)
```

```
    low_in, high_in = input_range
```

```
    if output_range is None:
```

```
        output_range = (0,1)
```

```
    low_out, high_out = output_range
```

```
    dst = (((src-low_in)/(high_in-low_in))**gamma)*(high_out-low_out)+low_out
```

```
    return dst
```

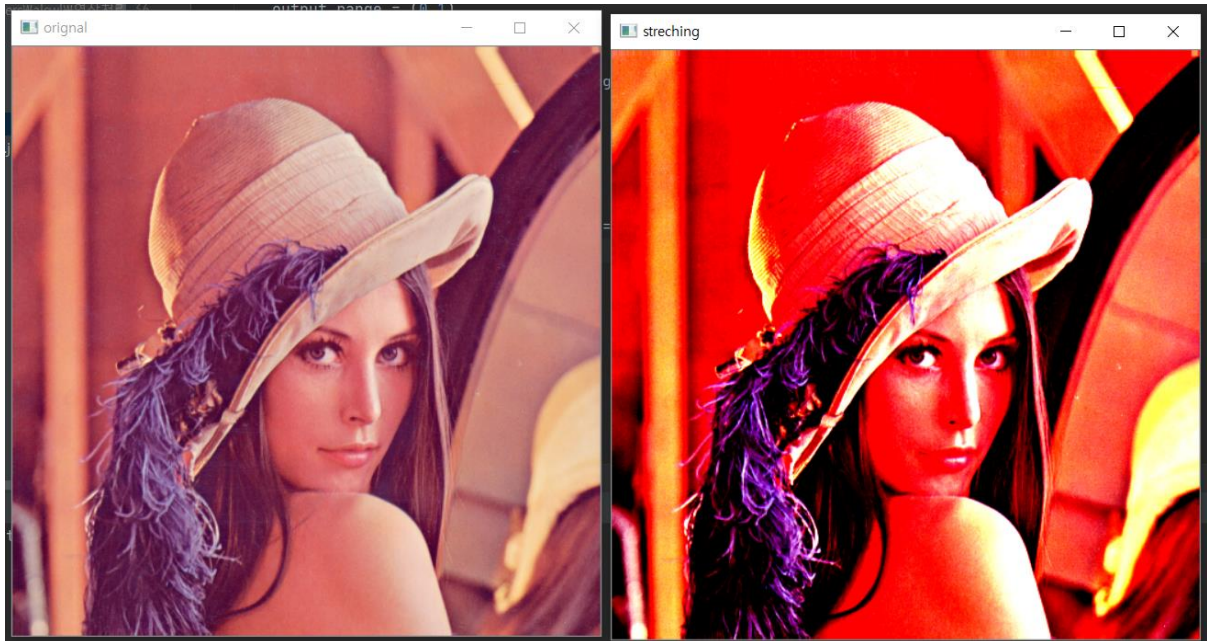
--- `imadjust` 함수를 생성합니다. Input과 output을 받아서 input은 `low_in`과 `high_in`에 넣어주고, output은 `low_out`과 `high_out`에 넣어줍니다.

연산과정을 거쳐 `dst`를 반환합니다.

```
img = img/255
```

```
dst = imadjust(img,[0.4, 0.7], [0, 1], gamma=1)
cv.imshow('original',img)
cv.imshow('stretching',dst)
```

--- input_range를 [0.4,0.7], output_range를 [0,1], gamma를 1로 두었을 때 영상을 확인하면



3채널 영상일 때는 이렇게 출력됩니다.

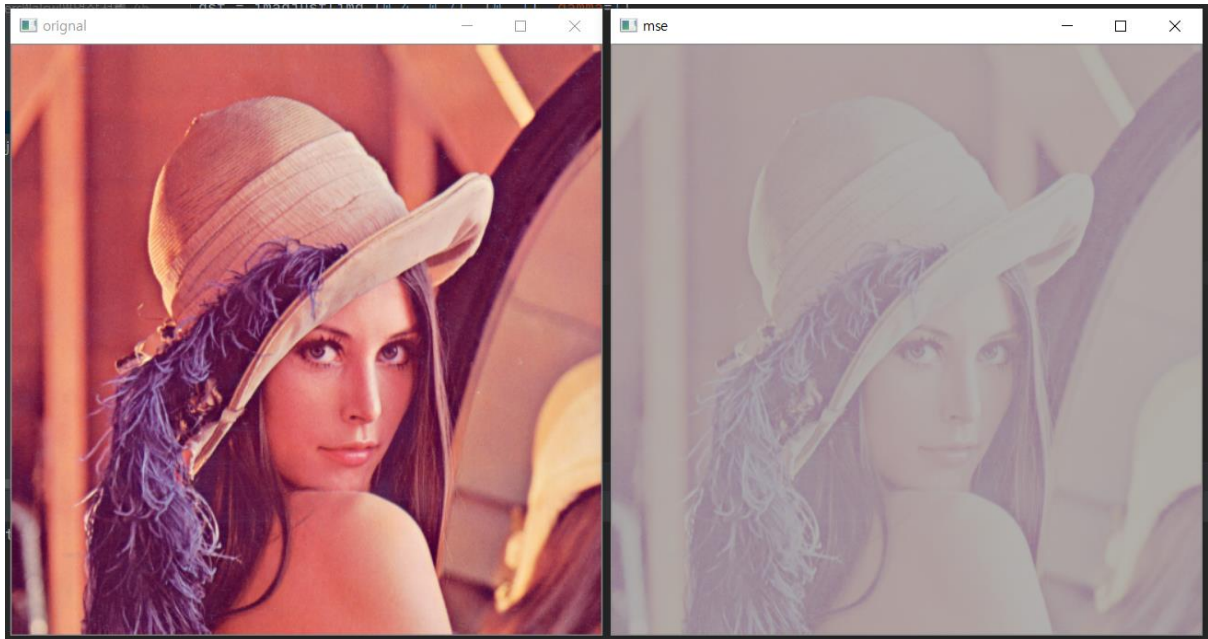
```
psnr=cv.PSNR(img, dst)
print(round(psnr,4))
```

--- psnr의 부동소수 4번째 값 까지 구하기 위해 round를 사용했습니다.

```
org=cv.imread(FullName)/255
img2=imadjust(org,[0, 1], [0.4, 0.7], 0.3)
cv.imshow('mse',img2)
mse=np.sum((org-img2)**2)

eps = np.finfo(mse).eps
print(format(mse,"10.3e"))
```

--- img2를 input_range를 [0, 1], output_range를 [0.4, 0.7], gamma를 0.3으로 설정했습니다.



실행시키면 이러한 영상이 출력됩니다.