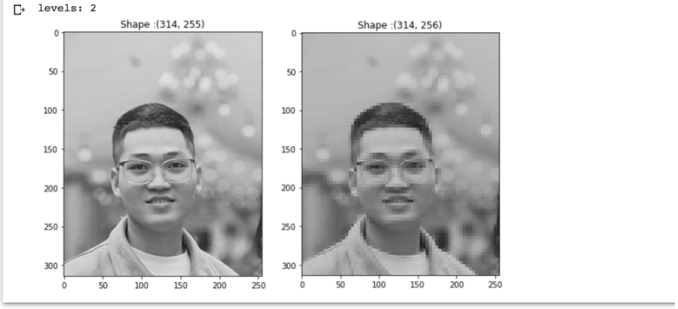
1311017-NguyenNgocBach-DIP-Lab06

Case1: low pass filter => so noise

```
import pywt, numpy
coeffs = pywt.wavedec2(original, 'db1', level = 2)
print("levels:", len(coeffs)-1)
cA2,(cH2, cV2, cD2),(cH1, cV1, cD1) = coeffs
cH2 = cH2 * 0
cD2 = cD2 * 0
cV2 = cV2 * 0
coeffs = cA2, (cH2, cV2, cD2), (cH1, cV1, cD1)
reconstruct_img = pywt.waverec2(coeffs, 'db1')
imshows([original, reconstruct_img],[Shape(original),Shape(reconstruct_img)],1,2,0)
original = gray
```

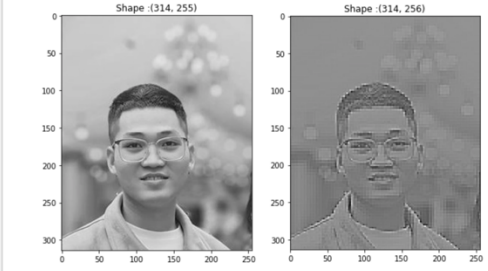
```
levels: 2
```



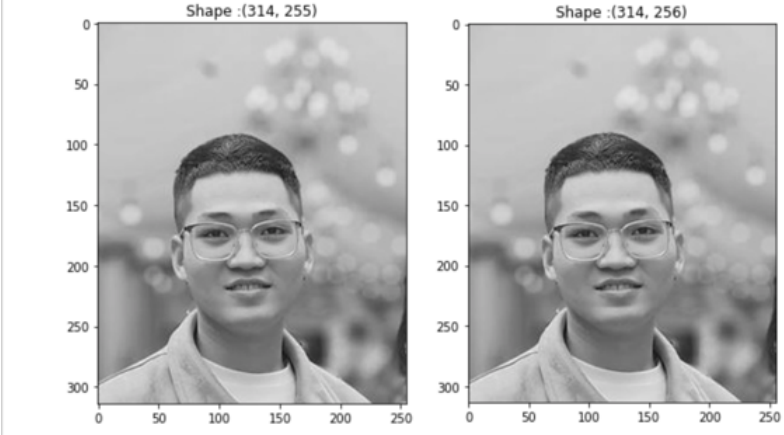
Case 2: high pass filter => low contrast

```
import pywt, numpy
coeffs = pywt.wavedec2(original, 'db1', level = 2)
print("levels:", len(coeffs)-1)
cA2,(cH2, cV2, cD2),(cH1, cV1, cD1) = coeffs
cH2 = cH2 * 5
cD2 = cD2 * 5
cV2 = cV2 * 5
coeffs = cA2, (cH2, cV2, cD2), (cH1, cV1, cD1)
reconstruct_img = pywt.waverec2(coeffs, 'db1')
imshows([original, reconstruct_img],[Shape(original),Shape(reconstruct_img)],1,2,0)
original = gray
```

```
levels: 2
```



Case 3: original => perfect

```
import pywt, numpy
coeffs = pywt.wavedec2(original, 'db1', level = 2)
print("levels:", len(coeffs)-1)
cA2,(cH2, cV2, cD2),(cH1, cV1, cD1) = coeffs
cH2 = cH2 * 1
cD2 = cD2 * 1
cV2 = cV2 * 1
coeffs = cA2, (cH2, cV2, cD2), (cH1, cV1, cD1)
reconstruct_img = pywt.waverec2(coeffs, 'db1')
imshows([original, reconstruct_img],[Shape(original),Shape(reconstruct_img)],1,2,0)
original = gray
```

```
levels: 2
```