

\* input gray level  $\rightarrow$  gr. piksel değeri.

$\rightarrow$  0 - 64 arasında "Log" dönüşüm ile 0.32 arası  
çerçeve ile aydınlatma bir görüntü sağlanır.

Log Transformatiyonu

$$s = c \log(1+r)$$

$\rightarrow$  gr. boyu.

0-255

$\downarrow$   
 $1+r \rightarrow$  256 değer.

Python Code:

$\rightarrow$  Open folder

$\rightarrow$  H04. (folder)

$\rightarrow$  Cimgviz  $\rightarrow$  folder.

$\rightarrow$  main.py

$\rightarrow$  terminal  $\Rightarrow$  python -m venv .\h04\gpi {environment}

$\rightarrow$  activate psd.

$\rightarrow$  pip install opencv-python

$\rightarrow$  pip install matplotlib.

main.py

$\rightarrow$

```

import cv2
import numpy as np

img_path = ". /images/image-name"
img = cv2.imread(img_path, 0)
print(img.shape) → 256, 256 boyutlu bir resim.

def log-trans(a c, c):
    s = c * log(1+r)
    s = c * np.log(1+r)
    return(s)

a = [5, 50, 150, 254, 255] ⇒ bu bir resim.

a-log = log-trans(a, 1)
print(a)
print(a-log)
→ a = np.array([5, 50, 150, 254, 255], dtype=np.uint8)

```

→ fourier-spectrum.  
 → tek kanal siyah beyaz olarak getir.  
 →  $\log$  fonksiyonu  
 →  $c$  katsayısı.  
 Affine form  
 →  $a$  data vericidir.

```

def log-trans(r, c):
    r = r.astype(np.float32) r = r.astype(np.float32)
    s = c * np.log(1+r)
    return s.astype(np.uint8)

q = np.array([5, 50, 150, 254, 255], dtype=np.float32)

```

~~print(a)~~  
~~print(a-b)~~

a-log = log-trans(a, 1)

print(a)

print(a-log)

0 se 5 arada parçalar.  
 Bu da donatılar olarak.

a-log = a-b - np.min(a-log)

print(a-log)

a-log = a-log / np.max(a-log)

print(a-b)

başlangıç 0'a ayar.

img-log = log-trans(img, 1)

plt.imshow(img-log, cmap="gray")

plt.show()

print(np.min(img-log)) → 0

print(np.max(img-log)) → 5

plt.imshow( ". / image.tif")

def im\_rescale(img):

```
img = img - np.min(img)
img = np.max(img)
img /= 255
return img
```

def log\_transform(r,c):  
r = r.astype(np.float32)  
s = c \* np.log(1 + r)  
s = im\_rescale(s)  
return s.astype(np.uint8)

hstacked = ~~np.hstack~~  
np.hstack((img, img-log))

plt.imshow(hstacked)

→ koyu olan kısımlar

