

# **Pengolahan Citra Digital**

Tugas Ke-2



Nama : Ulinnuha Luthfi

NIM : 5301414063

Dosen : Alfa Faridh Suni S.T., M.T.

**PROGRAM STUDI PENDIDIKAN TEKNIK ELEKTRO**

**JURUSAN TEKNIK ELEKTRO FAKULTAS TEKNIK**

**UNIVERSITAS NEGERI SEMARANG**

**2017**

## SOAL

1. Buat Low Pass Filter
2. Buat High Pass Filter
3. Buat Histogram equalization

## JAWAB

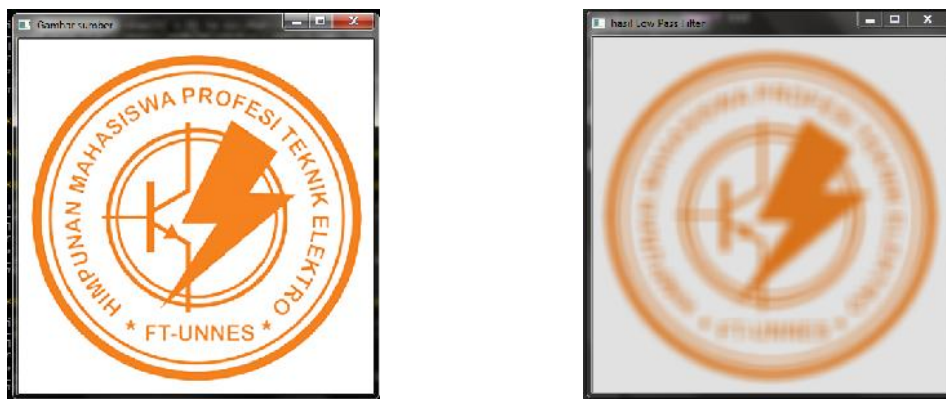
1. Low Pass Filter

```
• LPF.py - PYTHON CODE - Visual Studio Code
File Edit Selection View Go Debug Help

Welcome LPF.py HPF.py histogramequalization.py

1 import numpy as np
2 #import library numpy
3 import cv2
4 #import library openCV
5
6 scr = cv2.imread('HIMPROTE.png')
7 #deklarasi file sumber gambar
8 LPF = cv2.filter2D(scr,-1,np.ones((15,15),np.float32)/255)
9 #pemrosesan Low Pass Filter dengan mengisi digit satu(np.ones)
10 #dengan kernel 15x15
11 #>kernel>pula tingkat filter rendahnya (semakin blur gambarnya)
12
13 cv2.imshow('Gambar sumber',scr)
14 #menampilkan gambar asli
15 cv2.imshow('hasil Low Pass Filter',LPF)
16 #menampilkan gambar hasil Low Pass Filter
17
18 cv2.waitKey()
19 #menunggu input dari keyboard
20 cv2.destroyAllWindows()
21 #lalu tutup semua windows jika ada keyboard yang ditekan
22
```

Gambar 1. Sketch Low Pass Filter



Gambar 2. Gambar Asli (kiri) Gambar Hasil Low Pass Filter (kanan)

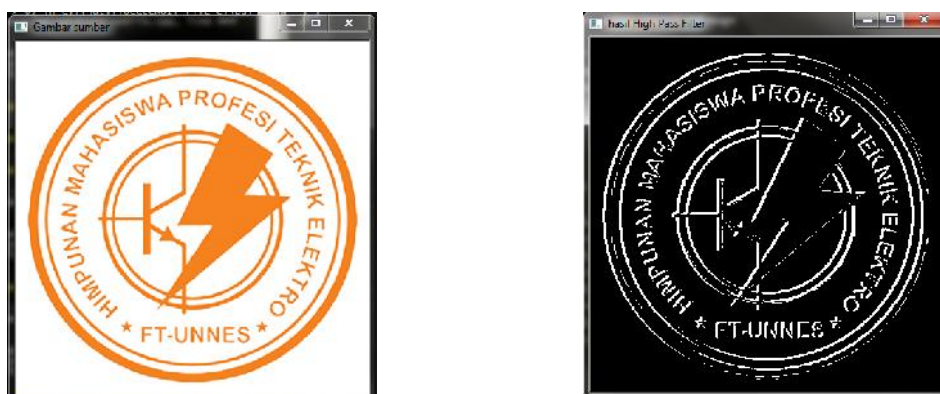
## 2. High Pass Filter

```
HPF.py - PYTHON CODE - Visual Studio Code
File Edit Selection View Go Debug Help

Welcome LPF.py HPF.py histogramequalization.py

1 import numpy as np
2 #import library numpy
3 import cv2
4 #import library openCV
5 from scipy import ndimage as im
6 #import library ndimage, pemanggilan fungsi dengan "im"
7
8 scr = cv2.imread('HIMPROTE.png')
9 #deklarasi file sumber gambar
10 gray= cv2.cvtColor(scr, cv2.COLOR_BGR2GRAY)
11 #konvert file sumber (RGB) ke grayscale
12 data= np.array(gray, dtype=float)
13 #pemanggilan perintah matriks dari kernel
14
15 kernel = np.array([[ -30, 7, -7],
16                    [ 2, 0, 30],
17                    [ -3, 1, -1]])
18 #data matriks
19
20 HPF = im.convolve(data, kernel)
21 #pemrosesan High Pass Filter dengan cara
22 #mengkonvolusi matrik gray dengan kernel 3x3
23
24 cv2.imshow('Gambar sumber',scr)
25 #menampilkan gambar asli
26 cv2.imshow('hasil High Pass Filter',HPF)
27 #menampilkan gambar hasil Low Pass Filter
28
29 cv2.waitKey()
30 #menunggu input dari keyboard
31 cv2.destroyAllWindows()
32 #lalu tutup semua windows jika ada keyboard yang ditekan
```

Gambar 1. Sketch High Pass Filter



Gambar 2. Gambar Asli (kiri) Gambar Hasil High Pass Filter (kanan)

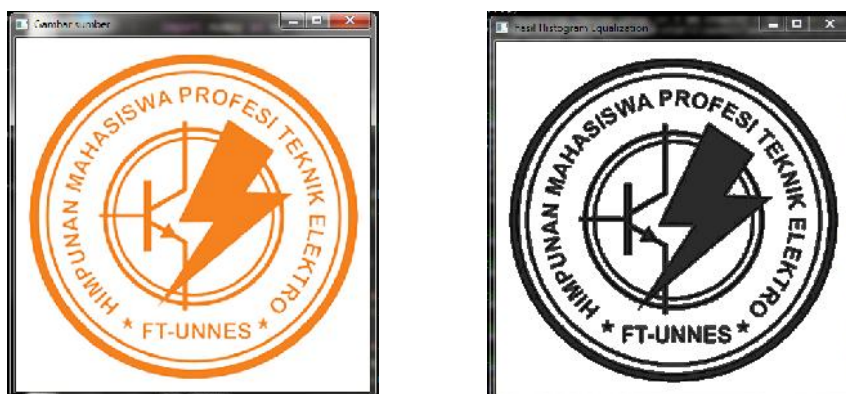
### 3. Histogram Equalization

```
histogramequalization.py - PYTHON CODE - Visual Studio Code
File Edit Selection View Go Debug Help

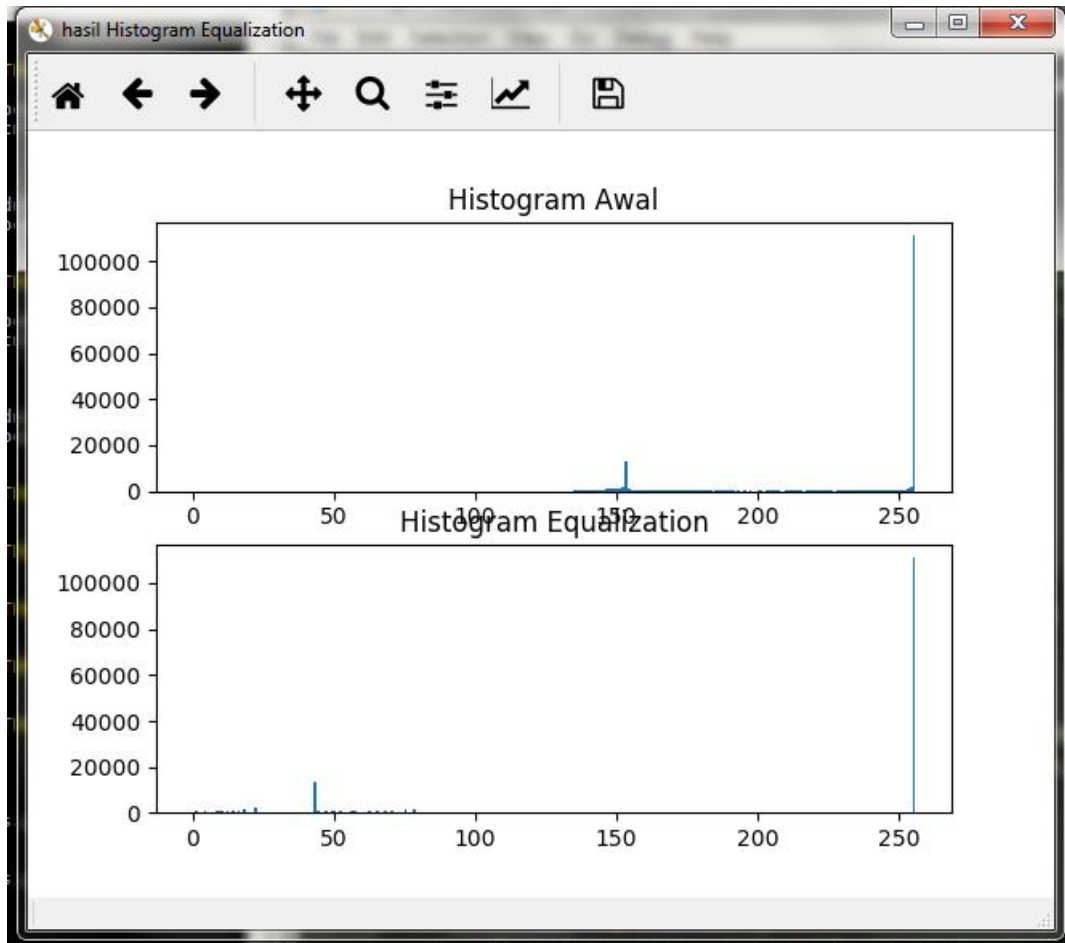
Welcome LPF.py HPF.py histogramequalization.py

1 import numpy as np
2 #import library numpy
3 import cv2
4 #import library openCV
5 from matplotlib import pyplot as plt
6 #import library pyplot, pemanggilan fungsi dengan "plt"
7 #untuk langsung plotting 2/ lebih histogram dalam satu window/ figure
8
9 scr = cv2.imread('HIMPROTE.png')
10 #deklarasi file sumber gambar
11 gray= cv2.cvtColor(scr, cv2.COLOR_BGR2GRAY)
12 #konvert file sumber (RGB) ke grayscale
13 equ = cv2.equalizeHist(gray)
14 #deklarasi fungsi histogram equalization
15
16 cv2.imshow('Gambar sumber',scr)
17 #menampilkan gambar asli
18 cv2.imshow('hasil Histogram Equalization',equ)
19 #menampilkan gambar hasil Low Pass Filter
20
21 plt.figure('hasil Histogram Equalization')
22 #memunculkan 1 figure dengan nama 'hasil Histogram Equalization'
23 plt.subplot(2,1,1),plt.hist(gray.ravel(),256,[0,256]),plt.title('Histogram Awal')
24 #plotting histogram awal di urutan ke-1 dalam figure
25 plt.subplot(2,1,2),plt.hist(equ.ravel(),256,[0,256]), plt.title('Histogram Equalization')
26 #plotting histogram equalization di urutan ke-2 dalam figure
27 plt.show()
28 #memunculkan plot
29
30 cv2.waitKey()
31 #menunggu input dari keyboard
32 cv2.destroyAllWindows()
33 #lalu tutup semua windows jika ada keyboard yang ditekan
```

Gambar 1. Sketch Histogram Equalization



Gambar 2. Gambar Asli (kiri) Gambar Hasil Histogram Equalization (kanan)



Gambar 3. Bentuk Histogram Equalization Dalam Grafik