

PENGOLAHAN CITRA DIGITAL

Low Pass Filter, High Pass Filter dan Histogram



Nama : Dita Puspitasari
NIM : 5301414011
Rombel : 01
Prodi : Pendidikan Teknik Elektro, S1

PENDIDIKAN TEKNIK ELEKTRO FAKULTAS TEKNIK

UNIVERSITAS NEGERI SEMARANG

2017

Low Pass Filter

Low Pass Filter digunakan untuk meneruskan sinyal berfrekuensi rendah dan meredam sinyal berfrekuensi tinggi. Pada pengolahan citra digital low pass filter menghasilkan gambar yang lebih halus dan cenderung blur.

Syntax dari Low Pass Filter

```
1 import numpy as np
2 import cv2
3
4
5 img = cv2.imread('rose.jpg')
6 lpf = cv2.filter2D(img,-1,np.ones((5,5),np.float32)/25) #membuat low pass filter dengan kernel 5x5
7
8 cv2.imshow('Gambar Asli',img)
9 cv2.imshow('Low Pass Filter',lpf)
10
11
12 cv2.waitKey()
13 cv2.destroyAllWindows()
14
```

`img = cv2.imread('rose.jpg')` *gambar yang dibaca adalah 'rose.jpg'*
`lpf = cv2.filter2D(img,-1,np.ones((5,5),np.float32)/25)` *membuat low pass filter dengan kernel 5x5.*

Gambar yang akan ditampilkan yaitu gambar asli dan gambar yang telah di low pass filter.

Gambar Asli



Gambar setelah di Low Pass Filter



High Pass Filter adalah jenis filter yang melewatkan frekuensi tinggi tetapi mengurangi amplitude frekuensi yang lebih rendah daripada frekuensi cutoff. Pada pengolahan citra digital high pass filter menghasilkan gambar yang lebih kasar dan terkesan tajam.

Syntax dari High Pass Filter

```
1  import numpy as np
2  import cv2
3  from scipy import ndimage
4
5
6
7  img = cv2.imread('rose.jpg')
8  gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
9  data = np.array(gray, dtype=float)
10
11
12  kernel = np.array([[ -9,  9, -9, ],
13                    [  9,  0,  9, ],
14                    [ -9,  9, -9, ]])
15  highpass_5x5 = ndimage.convolve(data, kernel)
16
17  cv2.imshow('Gambar Asli',img)
18  cv2.imshow('High Pass Filter',highpass_5x5)
19
20  cv2.waitKey(0)
21  cv2.destroyAllWindows()
```

Kernel yang digunakan pada High Pass Filter adalah 5x5

Gambar Asli



Gambar setelah di High Pass Filter



Histogram

Histogram adalah tampilan grafis dari tabulasi frekuensi yang digambarkan dengan grafis batangan sebagai manifestasi data binning.

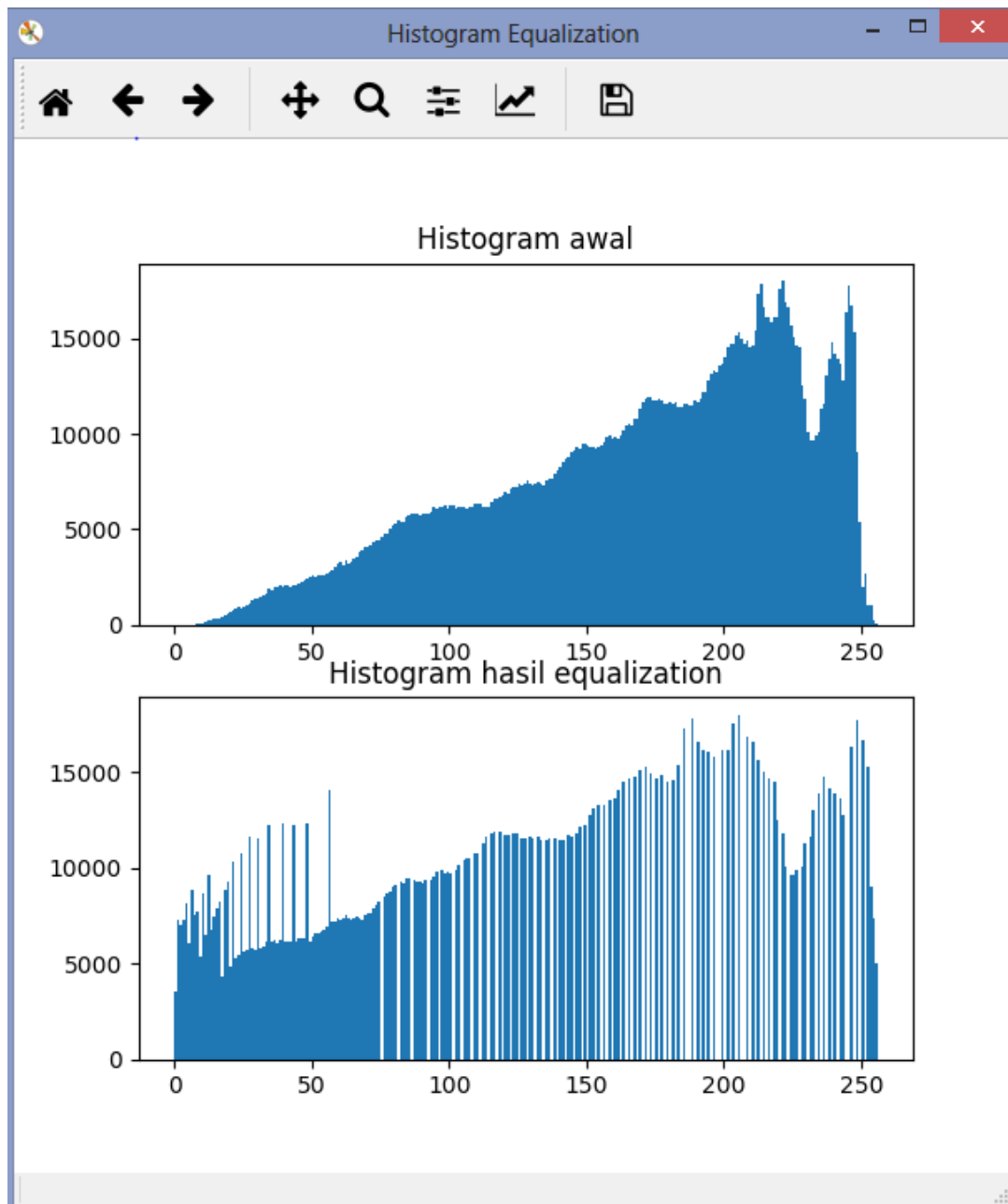
Syntax Histogram

```
1 import cv2
2 import numpy as np
3 from matplotlib import pyplot as plt #Matplotlib hadir dengan fungsi merencanakan histogram: matplotlib.pyplot.hist ().
4 #Ini langsung menemukan histogram dan plot itu. Anda tidak perlu menggunakan fungsi calcHist () atau np.histogram ()
5 #untuk menemukan histogram.
6
7 img = cv2.imread('rose.jpg')
8 gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
9
10 equ = cv2.equalizeHist(gray) #rumus equalization
11
12 cv2.imshow('Gambar Asli',gray)
13 cv2.imshow('Histogram Equalization', equ)
14
15 plt.figure('Histogram Equalization')
16 plt.subplot(2,1,1),plt.hist(gray.ravel(),256,[0,256]),plt.title('Histogram awal')
17 plt.subplot(2,1,2),plt.hist(equ.ravel(),256,[0,256]),plt.title('Histogram hasil equalization')
18 plt.show()
19
20 cv2.waitKey(0)
21 cv2.destroyAllWindows()
```

Gambar Asli



Hasil Histogram



Nilai histogram awal dimulai dari angka yang lebih besar dari 0 atau tidak tepat pada 0. Sedangkan hasil histogram equalization nilai histogram dimulai dari nilai 0 sampai sekitar 255.

Perubahan gambar agar histogram yang dihasilkan dimulai dari nilai 0 sampai dengan 255

