

Pengolahan Citra Digital

Tugas Ke-2 Opencv



Nama : Dewangga Pratama
NIM : 5301414088
Dosen : Dr. Hari Wibawanto, M.t.
Kuntoro Adi Nugroho, S.t. M.eng.

PROGRAM STUDI PENDIDIKAN TEKNIK ELEKTRO
JURUSAN TEKNIK ELEKTRO FAKULTAS TEKNIK
UNIVERSITAS NEGERI SEMARANG

2017

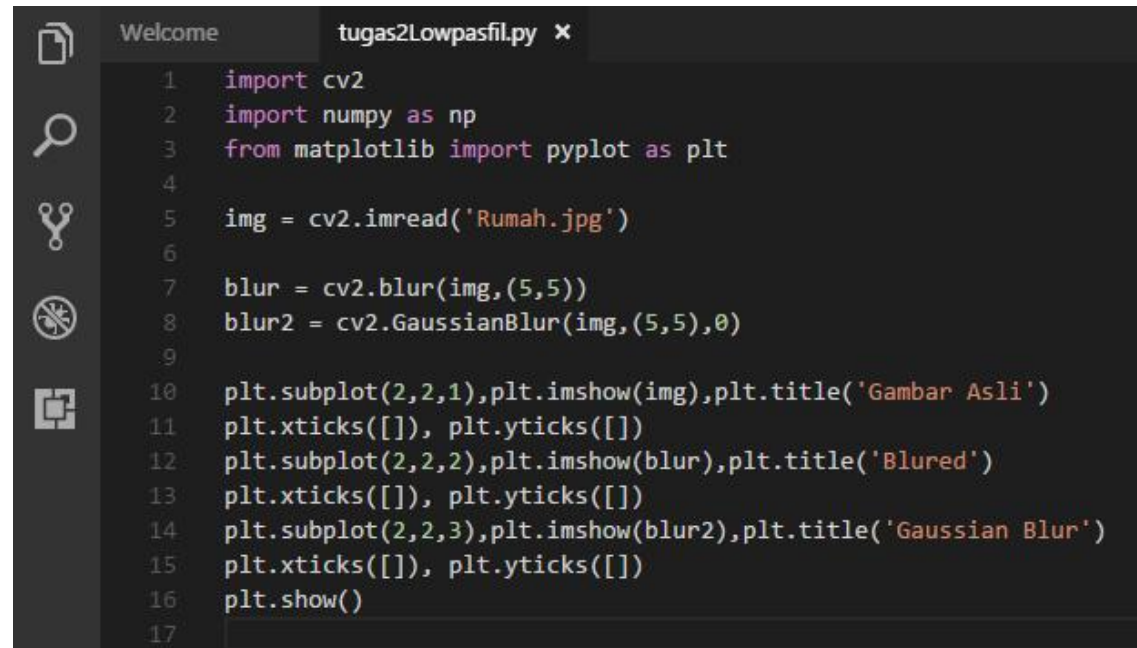
SOAL

Buatlah filter image menggunakan Low Pass Filter dan High Pass Filter kemudian buat Histogram dari hasil filter tersebut!

JAWAB

Coding :

LOWPASS FILTER



```
1 import cv2
2 import numpy as np
3 from matplotlib import pyplot as plt
4
5 img = cv2.imread('Rumah.jpg')
6
7 blur = cv2.blur(img,(5,5))
8 blur2 = cv2.GaussianBlur(img,(5,5),0)
9
10 plt.subplot(2,2,1),plt.imshow(img),plt.title('Gambar Asli')
11 plt.xticks([], plt.yticks([]))
12 plt.subplot(2,2,2),plt.imshow(blur),plt.title('Blured')
13 plt.xticks([], plt.yticks([]))
14 plt.subplot(2,2,3),plt.imshow(blur2),plt.title('Gaussian Blur')
15 plt.xticks([], plt.yticks([]))
16 plt.show()
17
```

Penjelasan :

- Line 1-3 :

```
import cv2
import numpy as np
from matplotlib import pyplot as plt
```

(Digunakan untuk mengimpor modul atau library yang akan digunakan)

- Line 5 :

```
img = cv2.imread('Rumah.jpg')
```

Melakukan inisialisasi pada gambar. Gambar yang akan digunakan diletakkan dalam satu folder dengan file python.

- Line 7 :

```
blur = cv2.blur(img,(5,5))
```

Menggunakan fungsi cv2.blur() untuk mengganti setiap nilai pixel didalam sebuah gambar dengan rata-rata nilai dari pixel tetanggannya, dan mencakup dirinya sendiri

- Line 8 :

```
blur2 = cv2.GaussianBlur(img,(5,5),0)
```

Menggunakan fungsi cv2.GaussianBlur(). Dalam teori Gaussian, setiap titik dari gambar akan dihitung tidak-nol, sehingga seluruh gambar harus dihitung setiap pixelnya. (5,5) adalah ukuran kernel lebar dan tinggi

- Line 10-11 :

```
plt.subplot(2,2,1),plt.imshow(img),plt.title('Gambar Asli')
plt.xticks([], plt.yticks([]))
```

Digunakan untuk menampilkan gambar hasil kedalam satu figure(Gambar Asli)

- Line 12-13 :

```
plt.subplot(2,2,2),plt.imshow(blur),plt.title('Blured')
plt.xticks([], plt.yticks([]))
```

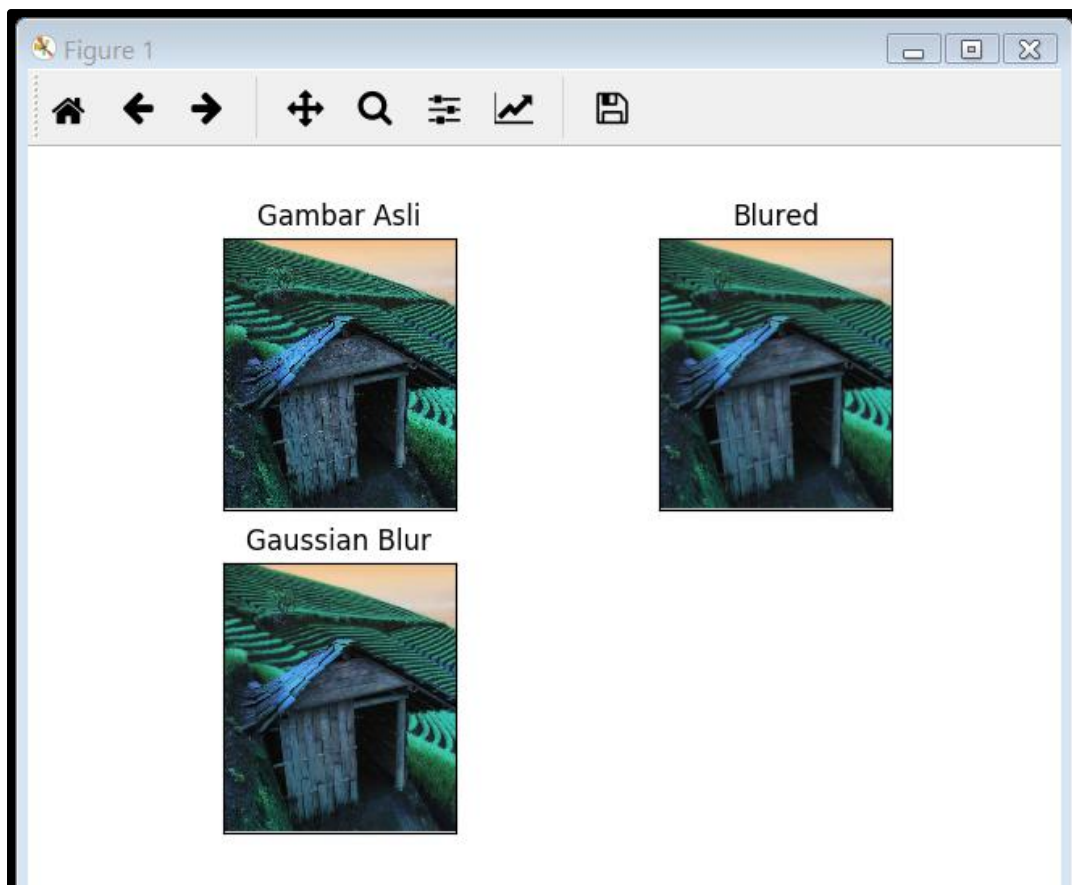
Digunakan untuk menampilkan gambar hasil kedalam satu figure(Blured)

- Line 14-16 :

```
plt.subplot(2,2,3),plt.imshow(blur2),plt.title('Gaussian Blur')
plt.xticks([], plt.yticks([]))
plt.show()
```

Digunakan untuk menampilkan gambar hasil kedalam satu figure(GaussianBlure) Dan menampilkan semua gambar pada Figure

HASIL GAMBAR :



Coding :

HIGHPASS FILTER

```
Welcome tugas2Highpasfil.py •
1  import cv2
2  import numpy as np
3  from matplotlib import pyplot as plt
4
5  # loading image
6  #img0 = cv2.imread('SanFrancisco.jpg')
7  img0 = cv2.imread('Bunga.jpg',)
8
9  # converting to gray scale
10 gray = cv2.cvtColor(img0, cv2.COLOR_BGR2GRAY)
11
12 # remove noise
13 img = cv2.GaussianBlur(gray,(3,3),0)
14
15 # convolute with proper kernels
16 laplacian = cv2.Laplacian(img, cv2.CV_64F)
17 sobelx = cv2.Sobel(img,cv2.CV_64F,1,0,ksize=5)
18 sobely = cv2.Sobel(img,cv2.CV_64F,0,1,ksize=5)
19
20 plt.subplot(2,2,1), plt.imshow(img,cmap = 'gray')
21 plt.title('Original'), plt.xticks([], plt.yticks([]))
22 plt.subplot(2,2,2), plt.imshow(laplacian,cmap = 'gray')
23 plt.title('Laplacian'), plt.xticks([], plt.yticks([]))
24 plt.subplot(2,2,3), plt.imshow(sobelx, cmap = 'gray')
25 plt.title('Sobel X'), plt.xticks([], plt.yticks([]))
26 plt.subplot(2,2,4), plt.imshow(sobely,cmap = 'gray')
27 plt.title('Sobel Y'), plt.xticks([], plt.yticks([]))
28 plt.show()
```

Penjelasan :

- Line 1-3 :

```
import cv2
import numpy as np
from matplotlib import pyplot as plt
```

(Digunakan untuk mengimpor modul atau library yang akan digunakan)

- Line 7 :

```
img0 = cv2.imread('Bunga.jpg',)
```

Digunakan untuk menginisialisasi gambar.

- Line 10:

```
gray = cv2.cvtColor(img0, cv2.COLOR_BGR2GRAY)
```

Digunakan untuk menginisialisasi gambar dan merubah gambar warna menjadi abu-abu
- Line 13:

```
img = cv2.GaussianBlur(gray,(3,3),0)
```

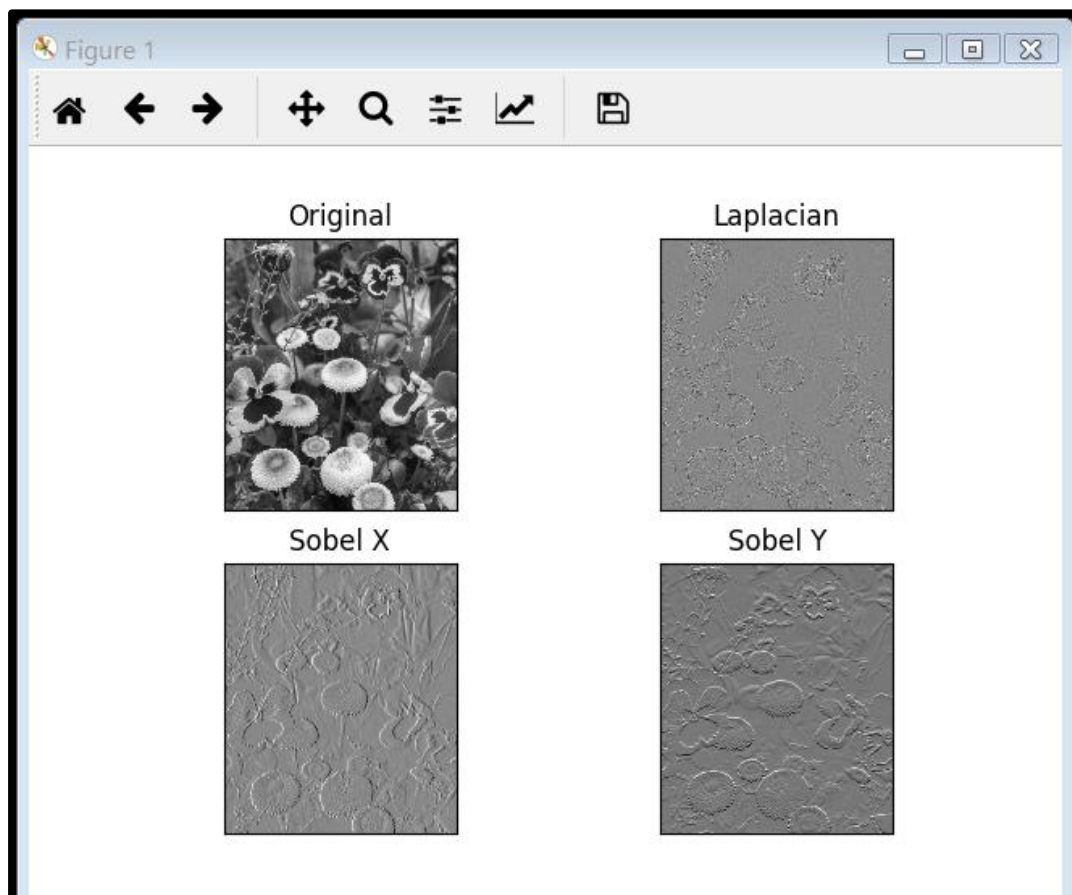
Digunakan menginisialisai gambar dan merubahnya menjadi efek kabur Gaussian, karena efek gaussian sangat baik digunakan untuk menghilangkan noise.
- Line 16-18 :

```
laplacian = cv2.Laplacian(img, cv2.CV_64F)
sobelx = cv2.Sobel(img,cv2.CV_64F,1,0,ksize=5)
sobely = cv2.Sobel(img,cv2.CV_64F,0,1,ksize=5)
```

Digunakan untuk mengkonvolusi gambar dengan kernel yang tepat.
Efek laplacian, cv2.CV_64F digunakan untuk meminta ukuran gambar tujuan.
Efek sobelx menginisialisasi gambar menjadi efek sobel x-dir, dan
Efek sobely gambar di inisialisasi menjadi efek sobel y-dir terhadap axes Y
- Line 20-28 :

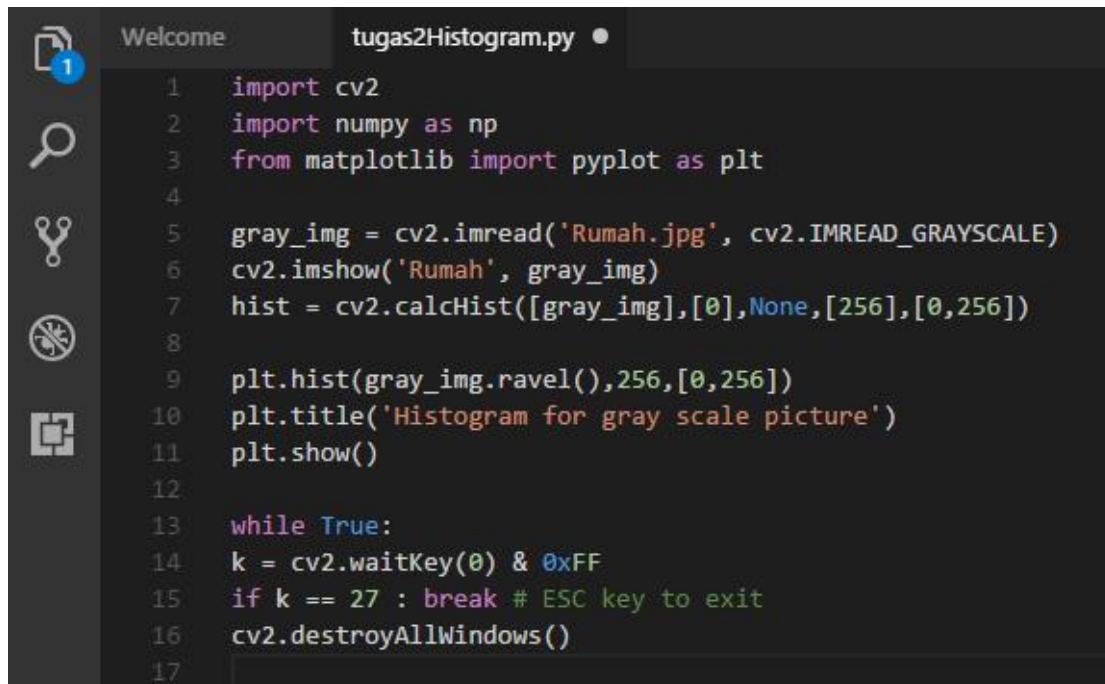
Digunakan untuk menampilkan hasil gambar kedalam satu figure.

HASIL GAMBAR :



Coding :

Histogram



```
1 import cv2
2 import numpy as np
3 from matplotlib import pyplot as plt
4
5 gray_img = cv2.imread('Rumah.jpg', cv2.IMREAD_GRAYSCALE)
6 cv2.imshow('Rumah', gray_img)
7 hist = cv2.calcHist([gray_img],[0],None,[256],[0,256])
8
9 plt.hist(gray_img.ravel(),256,[0,256])
10 plt.title('Histogram for gray scale picture')
11 plt.show()
12
13 while True:
14     k = cv2.waitKey(0) & 0xFF
15     if k == 27 : break # ESC key to exit
16     cv2.destroyAllWindows()
17
```

Penjelasan :

- Line 1-3 :

```
import cv2
import numpy as np
from matplotlib import pyplot as plt
```

(Digunakan untuk mengimpor modul atau library yang akan digunakan)

- Line 5-6 :

```
gray_img = cv2.imread('Rumah.jpg', cv2.IMREAD_GRAYSCALE)
cv2.imshow('Rumah', gray_img)
```

Digunakan untuk membaca gambar dan merubahnya menjadi gambar grayscale. Kemudian menampilkan nya kedalam figure

- Line 7:

```
hist = cv2.calcHist([gray_img],[0],None,[256],[0,256])
```

cv2.calcHist() adalah fungsi untuk histogram.[gray_img], gambar yang digunakan adalah type uint8 atau folat32.

1. [0] digunakan karena input gambar adalah grayscale, jika menggunakan gambar warna dapat menggunakan [0],[1] atau [2] untuk mengkalkulasi histogram channel biru, hijau dan merah, berturut-turut.

2. None, adalah mask image yang digunakan untuk menemukan histogram gambar penuh. [256] adalah histSize untuk skala penuh. [0,256] adalah range normal

- Line 9-11:

```
plt.hist(gray_img.ravel(),256,[0,256])  
plt.title('Histogram for gray scale picture')  
plt.show()
```

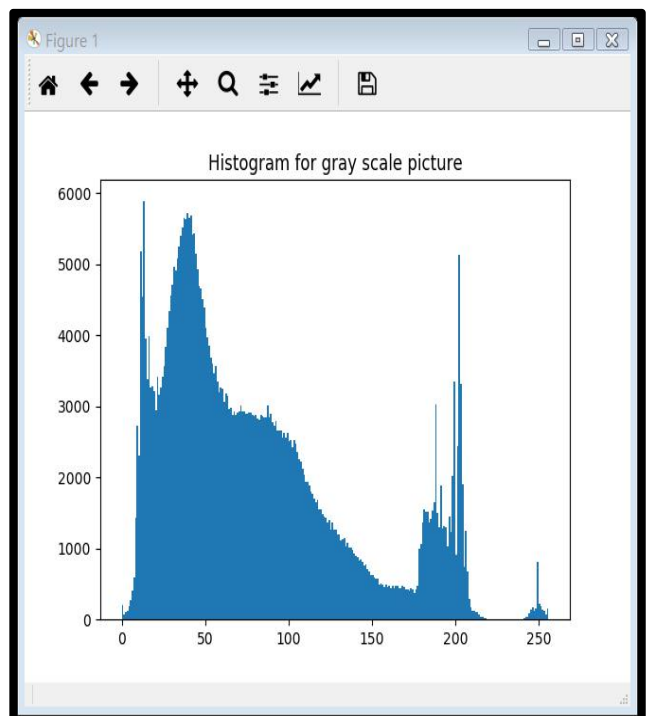
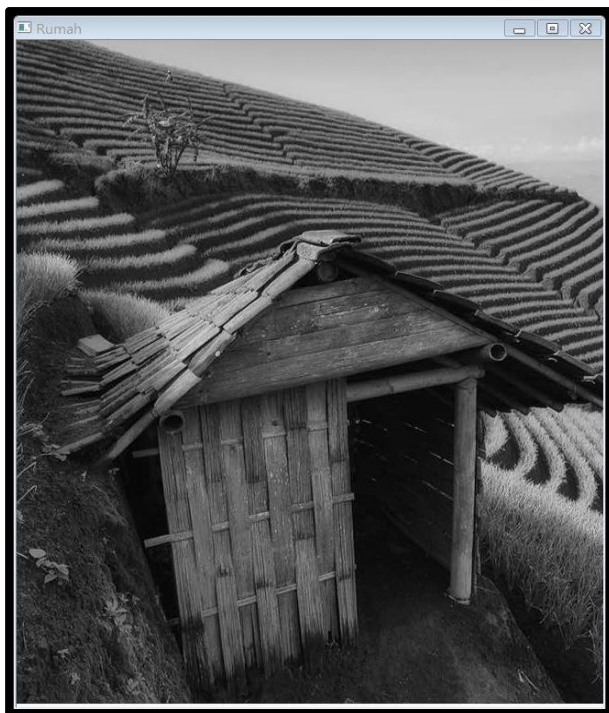
Digunakan untuk menampilkan gambar kedalam figure dengan nama :
'Histogram for gray scale picture'

- Line 13-16 :

```
while True:  
    k = cv2.waitKey(0) & 0xFF  
    if k == 27 : break # ESC key to exit  
    cv2.destroyAllWindows()
```

Perintah untuk menutup figure jika tombol ESC ditekan

HASIL GAMBAR :



NOTEBOOK :

Low pass filters (Smoothing)

Low pass filter (smoothing) Fungsi low pass filter saya gunakan untuk mengaburkan gambar (blurring) dengan efek Average dan Gaussian, bekerja dengan menghapus ruang derau frekuensi tinggi dari gambar digital. Low pass filters digunakan untuk menggeser jendela penghubung antara efek pixel pertama dari gambar pada waktu yang sama, mengubah nilai dari suatu fungsi lokal region piksel. LPF (low pass filter) juga digunakan untuk membantu menghapus noises.

High pass filter

High pass filter digunakan untuk membuat gambar terlihat tajam. HPF bekerja sama dengan low pass filter, hanya saja yang berbeda ada pada konvolusi kernel. Salah satu penggunaan HPF adalah menemukan tepi dari sebuah gambar. HPF bekerja digunakan untuk menemukan tepi gambar. Edge detection (deteksi garis tepi) biasanya digunakan untuk mendeteksi tepi sebuah skema (sobel) penggunaan dasar deteksi tepi dan Laplacian.

Histogram

Histogram digunakan untuk merepresentasikan graphical dari distribusi intensitas gambar. Digunakan pula untuk mengkuantisasi nomor pixel tiap nilai intensitas.

