

Pengolahan Citra

Install python, IDE, opencv, menampilkan gambar

Dosen Pengampu

Hero Yudo Martono ST, MT



Disusun Oleh :

Nama : M. Faza Nur Husain

Nrp : 3121550004

**D3 PJJ AK TEKNIK INFORMATIKA
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA
TAHUN AKADEMIK 2021/2022**

1. Install python

```
Select Command Prompt - python

C:\Users\fazan>python
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

2. Install text editor



3. Install opencv menggunakan pip

```
Administrator: Command Prompt - python

C:\WINDOWS\system32>python
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import cv2
>>> print(cv2.__version__)
4.5.5
>>> _
```

4. Menampilkan gambar

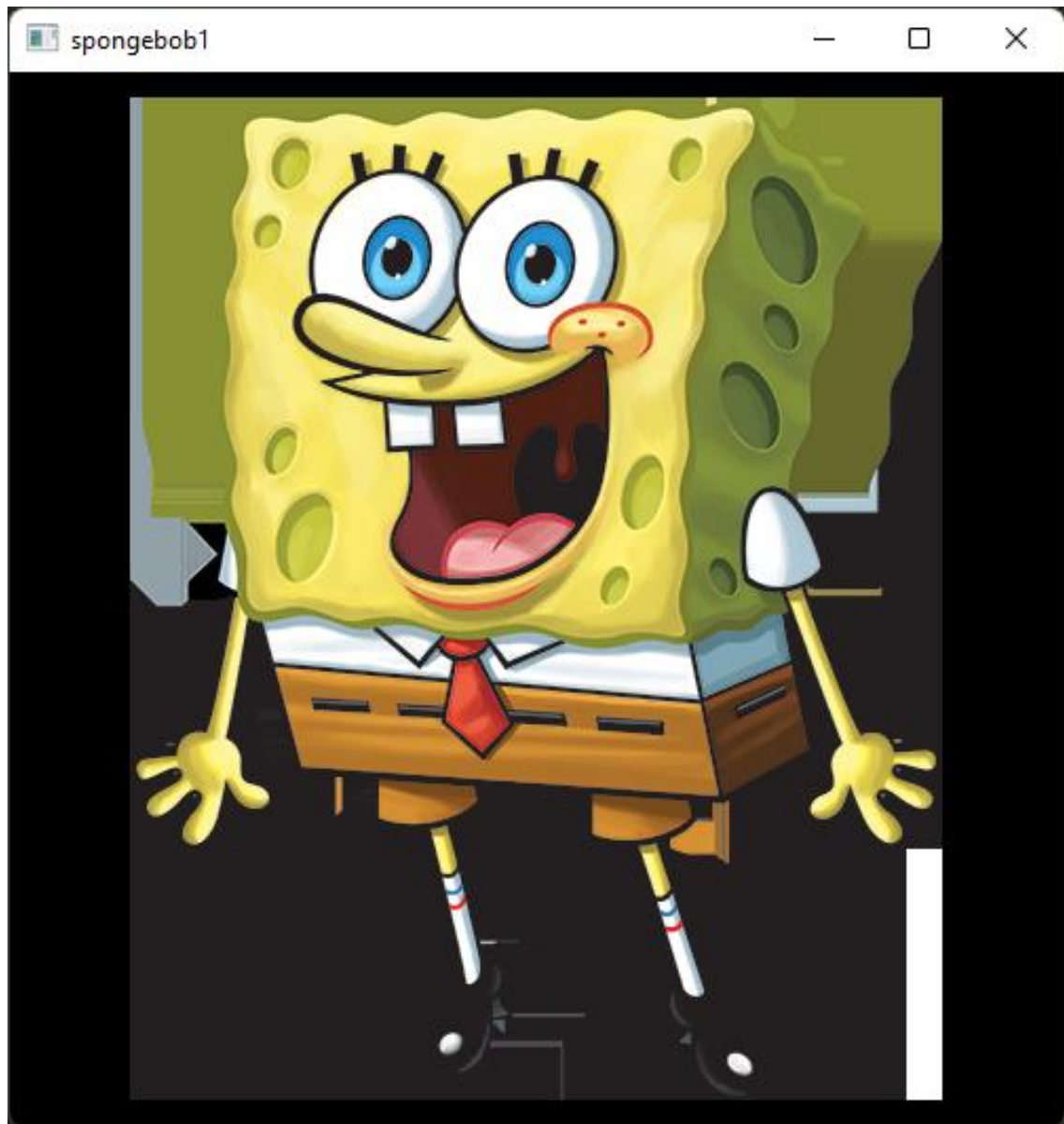
Sorce Code

```
import numpy as np
import cv2
from matplotlib import pyplot as plt
import matplotlib.image as mpimg
import time
from array import *

def load_image():
    img = cv2.imread('img/spongebob1.png')
    frame_resize = rescaleFrame(img, scale=5)
    cv2.imshow('arsip', img)
    cv2.imshow('arsip resize', frame_resize)
    cv2.waitKey(0)
    return

load_image()
```

Output Source Code



Pengolahan Citra
Pekan 02

Dosen Pengampu
Hero Yudo Martono ST, MT



Disusun Oleh :

Nama : M. Faza Nur Husain
Nrp : 3121550004

**D3 PJJ AK TEKNIK INFORMATIKA
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA
TAHUN AKADEMIK 2021/2022**

Tugas

Membuat gambar / mendapatkan nilai RGB per pixle, membuat gambar sederhana : garis, kotak, segitiga, lingkaran, memberi warna pada gambar. Konversi RGB gray biner dan flip gambar horizontal dan vertical, menggunakan fingsi dari library opencv

Source Code

```
from ast import Return
from cv2 import COLOR_BGRA2BGR
import numpy as np
import cv2
from matplotlib import pyplot as plt
import matplotlib.image as mpimg
import time
from array import *

def access_image():
    img01 = cv2.imread('img/sapi.jpg')
    row1, col1, n = img01.shape
    print(row1, col1)
    img02 = np.zeros((row1, col1, 3), np.uint8)
    img03 = np.zeros((140, 200, 3), np.uint8)
    img04 = np.zeros((140, 200, 3), np.uint8)

    img02 = cv2.cvtColor(img01, cv2.COLOR_BGR2RGB)
    img03 = img02.copy()

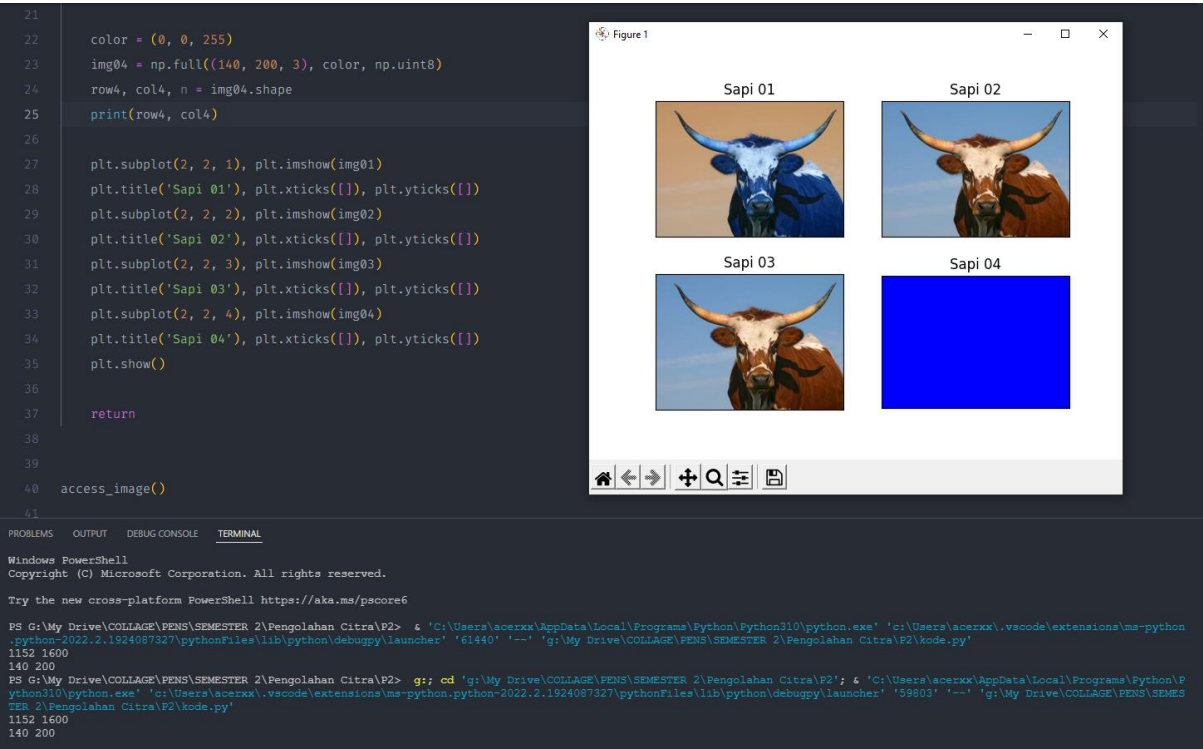
    color = (0, 0, 255)
    img04 = np.full((140, 200, 3), color, np.uint8)
    row4, col4, n = img04.shape
    print(row4, col4)

    plt.subplot(2, 2, 1), plt.imshow(img01)
    plt.title('Sapi 01'), plt.xticks([]), plt.yticks([])
    plt.subplot(2, 2, 2), plt.imshow(img02)
    plt.title('Sapi 02'), plt.xticks([]), plt.yticks([])
    plt.subplot(2, 2, 3), plt.imshow(img03)
    plt.title('Sapi 03'), plt.xticks([]), plt.yticks([])
    plt.subplot(2, 2, 4), plt.imshow(img04)
    plt.title('Sapi 04'), plt.xticks([]), plt.yticks([])
    plt.show()

    return

access_image()
```

Ouput Source Code:



Pengolahan Citra

Pekan 3. Enhancement : brigthness kuantisasi

Dosen Pengampu

Hero Yudo Martono ST, MT



Disusun Oleh :

Nama : M. Faza Nur Husain

Nrp : 3121550004

**D3 PJJ AK TEKNIK INFORMATIKA
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA
TAHUN AKADEMIK 2021/2022**

Source Code :

```

from sqlite3 import Row
from cv2 import imshow
import numpy as np
import cv2
from matplotlib import pyplot as plt
import matplotlib.image as mpimg
import time
from array import *

# Main Function

def flip_image():
    img = mpimg.imread('img/kuda.jpg')
    horizontal_img = cv2.flip(img, 1)
    vertical_img = cv2.flip(img, 0)
    both_img = cv2.flip(img, -1)

    plt.subplot(2, 2, 1), plt.imshow(img)
    plt.title('Original'), plt.xticks([]), plt.yticks([])
    plt.subplot(2, 2, 2), plt.imshow(horizontal_img)
    plt.title('Flip Horizontal'), plt.xticks([]), plt.yticks([])
    plt.subplot(2, 2, 3), plt.imshow(vertical_img)
    plt.title('Flip Vertikal'), plt.xticks([]), plt.yticks([])
    plt.subplot(2, 2, 4), plt.imshow(both_img)
    plt.title('Flip both'), plt.xticks([]), plt.yticks([])
    plt.show()
    return

def enchancement():
    img = mpimg.imread('img/kuda.jpg')
    row, col, n = img.shape
    img1 = np.zeros((row, col, 3), np.uint8)
    img2 = np.zeros((row, col, 3), np.uint8)
    img3 = np.zeros((row, col, 3), np.uint8)
    img4 = np.zeros((row, col, 3), np.uint8)
    img5 = np.zeros((row, col, 3), np.uint8)

    th = 50
    for y in range(0, col-1):
        for x in range(0, row-1):
            R, G, B = img[x, y]
            if (R+th) > 255:
                R = 255
            else:
                R = R+th
            if (G+th) > 255:
                G = 255
            else:
                G = G+th
            if (B+th) > 255:
                B = 255
            else:

```



```

        R = R+th
        img1[x, y] = [R, G, B]

th = 4
for y in range(0, col-1):
    for x in range(0, row-1):
        R, G, B = img[x, y]
        if (R*th) > 255:
            R = 255
        else:
            R = R*th
        if (G*th) > 255:
            G = 255
        else:
            G = G*th
        if (B*th) > 255:
            B = 255
        else:
            B = B*th
        img2[x, y] = [R, G, B]

xmax = 0
xmin = 300

for y in range(0, col-1):
    for x in range(0, row-1):
        R, G, B = img[x, y]
        gray = int((R+G+B)/3)
        if(gray > xmax):
            xmax = gray
        if(gray < xmin):
            xmin = gray

d = xmax-xmin
for y in range(0, col-1):
    for x in range(0, row-1):
        R, G, B = img[x, y]
        gray = int((R+G+B)/3)
        gray = int((255/d)*gray-xmin)
        img3[x, y] = [gray, gray, gray]

print("xmax=", xmax)
print("xmin=", xmin)

titles = ['Original Image', 'BRIGHTNESS', 'CONTRAST', 'AUTO SCALE']
images = [img, img1, img2, img3]
for i in range(4):
    plt.subplot(2, 2, i+1), plt.imshow(images[i], 'gray', vmin=0,
vmax=255)
    plt.title(titles[i])
    plt.xticks([], plt.yticks([]))
plt.show()
return

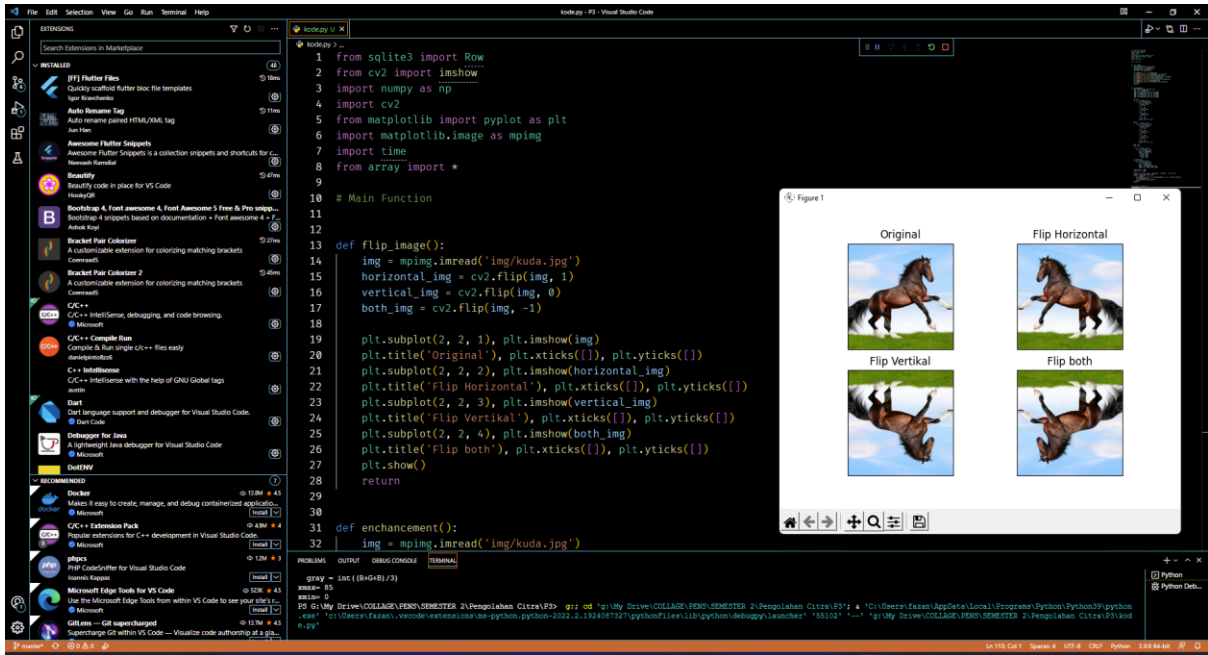
# =====Main Program=====

```

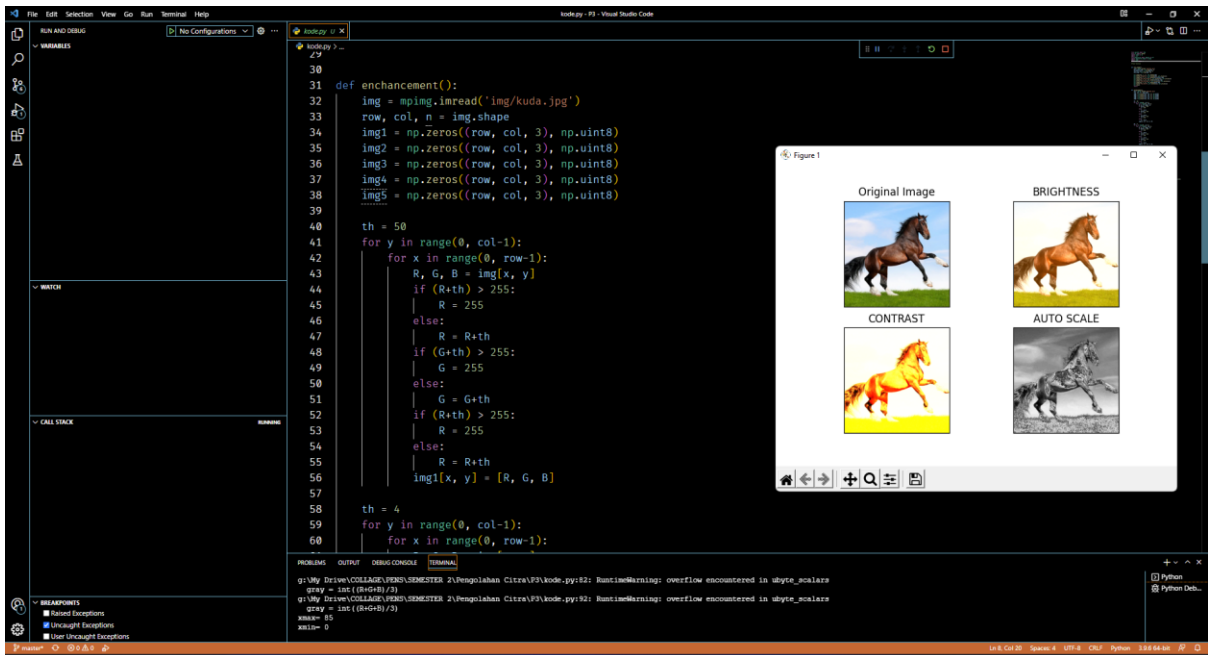
```
# flip_image()
# enhancement()
```

Output Code :

flip_image()



enhancement()



Pengolahan Citra

Pekan 4. Filtering LPH dan HPF in domain spatial

Dosen Pengampu

Hero Yudo Martono ST, MT



Disusun Oleh :

Nama : M. Faza Nur Husain

Nrp : 3121550004

**D3 PJJ AK TEKNIK INFORMATIKA
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA
TAHUN AKADEMIK 2021/2022**

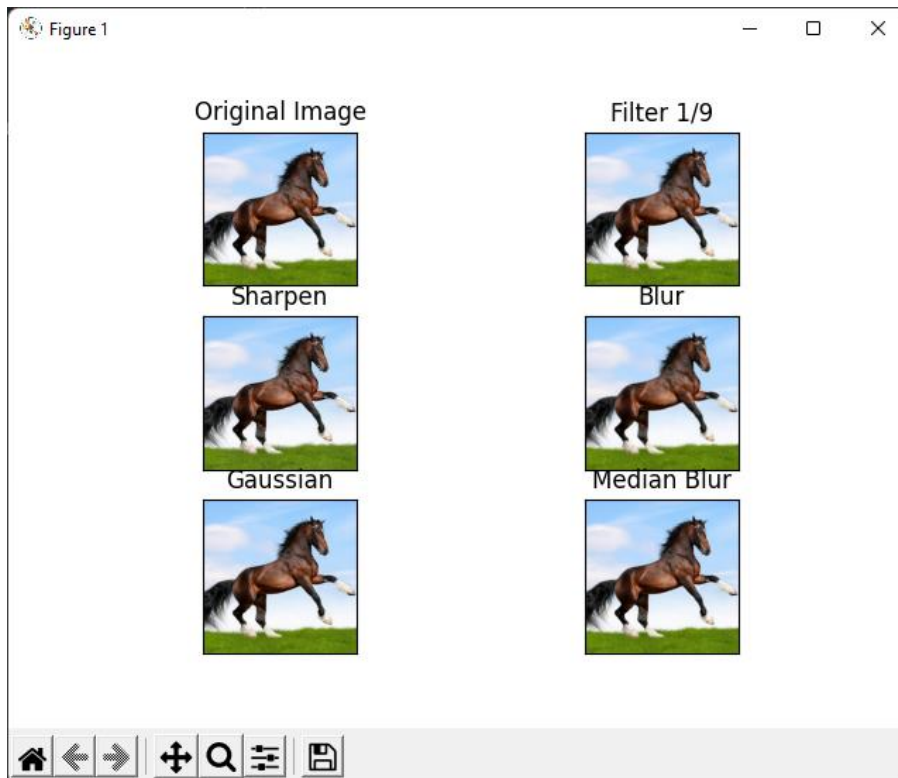
Source Code :

```
def convolution2D():
    img1 = cv2.imread('gambar/kuda.jpg')
    img1 = cv2.cvtColor(img1, cv2.COLOR_BGR2RGB)
    kernel = np.ones((3, 3), np.float32) / 9
    #print (kernel)
    img2 = cv2.filter2D(img1, -1, kernel)

    kernel = np.array([[0, -1, 0],
                       [-1, 5, -1],
                       [0, -1, 0]])
    img3 = cv2.filter2D(img1, -1, kernel)

    img4 = cv2.blur(img1, (5, 5))
    img5 = cv2.GaussianBlur(img1, (3, 3), 0)
    img6 = cv2.medianBlur(img1, 3)

    titles = ['Original Image', 'Filter 1/9',
              'Sharpen', 'Blur', 'Gaussian', 'Median Blur']
    images = [img1, img2, img3, img4, img5, img6]
    for i in range(6):
        plt.subplot(3, 2, i+1), plt.imshow(images[i], 'gray', vmin=0,
vmax=255)
        plt.title(titles[i])
        plt.xticks([], plt.yticks([]))
    plt.show()
    return
convolution2D()
```



Source Code :

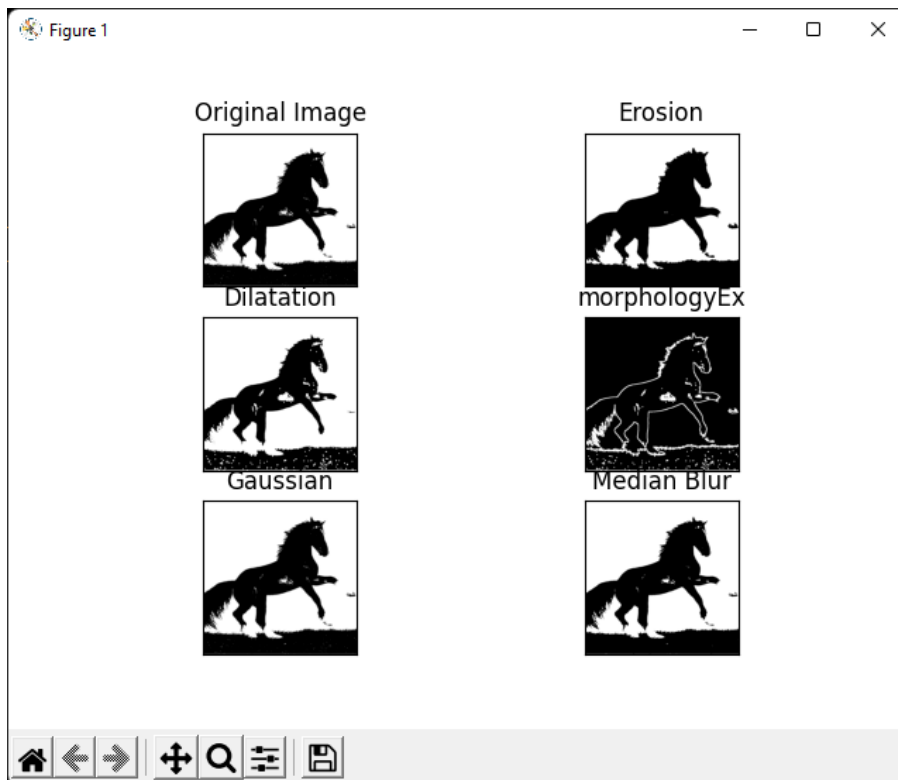
```
def dilatation():
    img1 = cv2.imread('gambar/kuda.jpg')

    # convert to black and white
    img1 = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)
    r, img1 = cv2.threshold(img1, 150, 255, cv2.THRESH_BINARY)
    # create kernel
    kernel = np.ones((5, 5), np.uint8)
    img2 = cv2.erode(img1, kernel)
    img3 = cv2.dilate(img1, kernel)
    img4 = cv2.morphologyEx(img1, cv2.MORPH_GRADIENT, kernel)

    img5 = cv2.GaussianBlur(img1, (3, 3), 0)
    img6 = cv2.medianBlur(img1, 3)

    titles = ['Original Image', 'Erosion', 'Dilatation',
              'morphologyEx', 'Gaussian', 'Median Blur']
    images = [img1, img2, img3, img4, img5, img6]
    for i in range(6):
        plt.subplot(3, 2, i+1), plt.imshow(images[i], 'gray', vmin=0,
vmax=255)
        plt.title(titles[i])
        plt.xticks([], plt.yticks([]))
    plt.show()
    return

dilatation()
```



Source Code :

```
def filtering():
    img1 = cv2.imread('gambar/kuda.jpg')
    kernel = np.array([[1, 1, 1, 1, 1],
                       [1, 1, 1, 1, 1],
                       [1, 1, 1, 1, 1],
                       [1, 1, 1, 1, 1],
                       [1, 1, 1, 1, 1]])

    kernel = kernel/25
    img2 = cv2.filter2D(img1, -1, kernel)
    kernel = np.array([[0.0, -1.0, 0.0],
                       [-1.0, 4.0, -1.0],
                       [0.0, -1.0, 0.0]])

    kernel = kernel/(np.sum(kernel) if np.sum(kernel) != 0 else 1)
    img3 = cv2.filter2D(img1, -1, kernel)
    kernel = np.array([[0.0, -1.0, 0.0],
                       [-1.0, 5.0, -1.0],
                       [0.0, -1.0, 0.0]])

    kernel = kernel/(np.sum(kernel) if np.sum(kernel) != 0 else 1)
    img4 = cv2.filter2D(img1, -1, kernel)

    # img4= cv2.morphologyEx(img1, cv2.MORPH_GRADIENT, kernel)
    # img5= cv2.GaussianBlur(img1, (3,3), 0)
    # img6= cv2.medianBlur(img1, 3)

    kernel = np.array([[ -1.0, -1.0, ],
                       [2.0, 2.0],
                       [ -1.0, -1.0]])

    kernel = kernel/(np.sum(kernel) if np.sum(kernel) != 0 else 1)
    img5 = cv2.filter2D(img1, -1, kernel)
    titles = ['original image', 'low pass', 'high pass',
              'high pass', 'cusom kernel', 'normal']
    images = [img1, img2, img3, img4, img5, img1]

    for i in range(6):
        plt.subplot(
            3, 2, i+1), plt.imshow(images[i], 'gray', vmin=-0, vmax=255)
        plt.title(titles[i])
        plt.xticks([], plt.yticks([]))
    plt.show()

    return
```

