**SOAL UAS PRAKTEK**

**PENGOLAHAN CITRA DIGITAL GENAP 2021-2022**

**KELAS 4D**

1. **Tes 1 : Smoothing**

**Code:**

from scipy import misc

import matplotlib.pyplot as plt

from skimage.filters import gaussian

foto = misc.face()

plt.gray()

plt.imshow(foto)

plt.show()

smoothed = gaussian(foto, *multichannel*=True, *sigma*=2)

plt.imshow(smoothed)

plt.show

Graphical user interface, text

Description automatically generated

Smoothing umumnya digunakan untuk menyamarkan noise pada citra dengan cara memberikan blur pada citra.

Hasil:

A raccoon eating a leaf

Description automatically generated with low confidenceA raccoon eating grass

Description automatically generated with low confidence

1. **Tes 2: Edge Detection**

**Jelaskan apa yang dimaksud dengan Edge Detection!**

Edge detection atau deteksi tepi adalah suatu proses yang menghasilkan tepi dari object gambar, tujuan dari edge detection adalah untuk menandai bagian yang menjadi detail gambar untuk memperbaiki detail dari gambar yang blur.

Untuk tes edge detection disini menggunakan metode sobel dengan mengimport melalui skimage untuk gambar yang telah dibuat menjadi berwarna abu abu sehingga detail tampak lebih jelas deteksi tepinya.

**Code:**

from skimage import data

import matplotlib.pyplot as plt

from skimage.filters import sobel

from skimage.color import rgb2gray

gambarasli = data.astronaut()

gambarabu = rgb2gray(gambarasli)

gambar\_edge = sobel(gambarabu)

plt.imshow(gambarabu, *cmap*=plt.cm.gray)

plt.show()

plt.imshow(gambar\_edge)

plt.show()

Text

Description automatically generated

A picture containing text, indoor, person, posing

Description automatically generatedA picture containing calendar

Description automatically generated

1. **Thresholding Local**

Jelaskan apa yang dimaksud dengan Local Thresholding!

teknik ambang batas lokal menggunakan nilai thresholding unik untuk subimage yang dipecah yang diperoleh dari seluruh gambar yang artinya metode ini gambar dipecah menjadi beberapa bagian gambar yang lebih kecil dan kemudian tiap bagian gambar dicari nilai thresholdingnya

**code:**

from skimage import data

import matplotlib.pyplot as plt

from skimage.filters import sobel

from skimage.color import rgb2gray

gambarasli = data.astronaut()

gambarabu = rgb2gray(gambarasli)

gambar\_edge = sobel(gambarabu)

plt.imshow(gambarabu, *cmap*=plt.cm.gray)

plt.show()

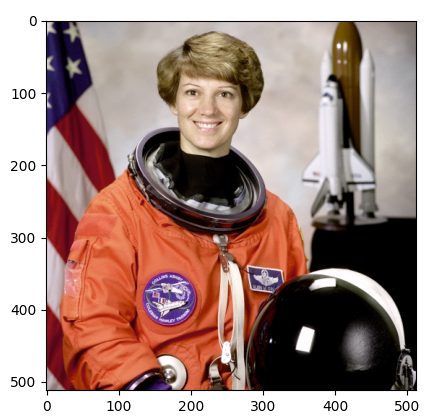
plt.gray()

plt.imshow(gambar\_edge)

plt.show()

Text

Description automatically generated

A picture containing text, book

Description automatically generated

1. **Thresholding Global**

**Jelaskan apa yang dimaksud dengan Global Thresholding!**

Global thresholding adalah teknik yang menggunakan nilai threshold tunggal untuk keseluruhan gambar, yang artinya nilai tunggal dari intensitas piksel digunakan untuk semua piksel dalam gambar untuk mengkonversi menjadi gambar binary. Hal ini dapat berfungsi dengan baik apabila semua gambar diambil dalam kondisi cahaya yang sama dan sering kali tidak benar.

**Code:**

from skimage.filters import try\_all\_threshold

from skimage import data

from skimage.color import rgb2gray

import matplotlib.pyplot as plt

gambarasli = data.astronaut()

gambarabu = rgb2gray(gambarasli)

plt.imshow(gambarasli)

plt.show()

fig, ax = try\_all\_threshold(

    gambarabu, *figsize*=(10, 8), *verbose*=False

)

fig.tight\_layout()

plt.show()

Text

Description automatically generated

A picture containing text, different, various, bunch

Description automatically generated

**5. Flip Image**

Buatlah tampilan output berupa citra flip vertical dan flip horizontal dari logo originalnya.

**Code:**

from PIL import Image

from matplotlib import pyplot as plt

original = Image.open("./logoqman.png")

flipvertical = original.transpose(*method*=Image.FLIP\_TOP\_BOTTOM)

flipvertical.save('flipvertical.png')

fliphorizontal =  original.transpose(*method*=Image.FLIP\_LEFT\_RIGHT)

fliphorizontal.save('fliphorizontal.png')

fig, axes = plt.subplots(1,3, *figsize*=(8,4))

ax = axes.ravel()

ax[0].imshow(original)

ax[0].set\_title("Original")

ax[1].imshow(flipvertical)

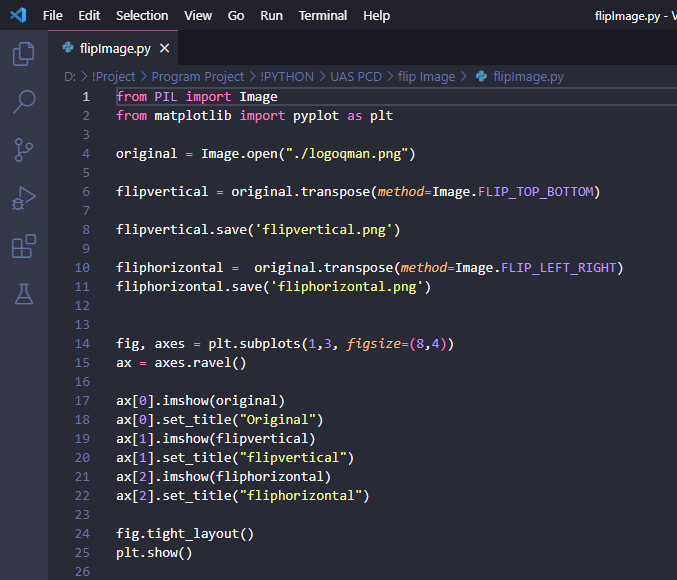
ax[1].set\_title("flipvertical")

ax[2].imshow(fliphorizontal)

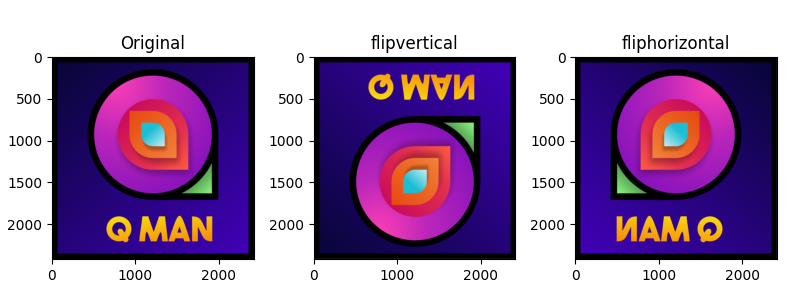
ax[2].set\_title("fliphorizontal")

fig.tight\_layout()

plt.show()



**Hasil**



Chart

Description automatically generated