

LAPORAN PRAKTIKUM

PEMROSESAN DATA MINGGU KE-5

Pengolahan Data Citra Menggunakan Matplotlib (Kuis)

Dosen Pengampu: Alfi Fadliana S.Si., M.Stat.



Oleh:

Bayu Kurniawan
(3322600019)

PROGRAM STUDI D4 SAINS DATA TERAPAN
DEPARTEMEN TEKNIK INFORMATIKA DAN KOMPUTER
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA

2023

Kuis Praktikum Pemrosesan Data

1. Cari perbedaan image processing menggunakan OpenCV dan Matplotlib

Jawab:

Tujuan Utama:

- ✓ OpenCV dirancang untuk pemrosesan gambar dan visi komputer dengan focus pada aplikasi yang lebih kompleks seperti deteksi wajah, pengenalan objek, dll.
- ✓ Matplotlib dirancang untuk visualisasi data dalam bentuk grafik dan plot, termasuk gambar.

Format gambar:

- ✓ OpenCV memuat gambar sebagai array NumPy dalam format BGR (blue, green, red).
- ✓ Matplotlib memuat gambar sebagai array NumPy dalam format RGB (red, green, blue).

Fungsi:

- ✓ OpenCV menawarkan fungsi pemrosesan gambar lanjutan seperti filter, thresholding, konversi ruang warna, deteksi tepi, pengenalan wajah, dll.
- ✓ Matplotlib menawarkan fungsi visualisasi seperti plot, histogram, scatter plot, dll.

Kecepatan:

- ✓ OpenCV dikenal sangat cepat dan efisien dalam pemrosesan gambar dan visi komputer karena dirancang khusus untuk itu.
- ✓ Matplotlib tidak secepat OpenCV karena fokus utamanya adalah visualisasi data dalam bentuk grafik dan plot.

Kesimpulan:

Apabila tujuan utama adalah pemrosesan pada data citra dan komputer vision, maka OpenCV adalah pilihan yang lebih baik. Namun, jika kita hanya perlu menampilkan gambar dan visualisasi data, maka Matplotlib akan menjadi pilihan yang lebih baik. Namun, keduanya dapat digunakan bersama-sama untuk memproses dan menampilkan gambar secara efisien dan efektif dalam Python.

2. Lakukan image processing menggunakan Matplotlib

Jawab:

```
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
from PIL import Image
img = np.asarray(Image.open("C:/Users/bayuk/OneDrive/Documents/AI/pens/smtr2/Pemrosesan Data/3/bayu.jpeg"))
print(repr(img))
```

Output exceeds the [size limit](#). Open the full output data [in a text editor](#)

```
array([[ 50,  64,  67],
       [ 50,  64,  67],
       [ 50,  64,  67],
       ...,
       [ 25,  31,  31],
       [ 27,  33,  33],
       [ 29,  35,  35]],

       [[ 50,  64,  67],
       [ 50,  64,  67],
       [ 50,  64,  67],
       ...,
       [ 24,  30,  30],
       [ 26,  32,  32],
       [ 28,  34,  34]],

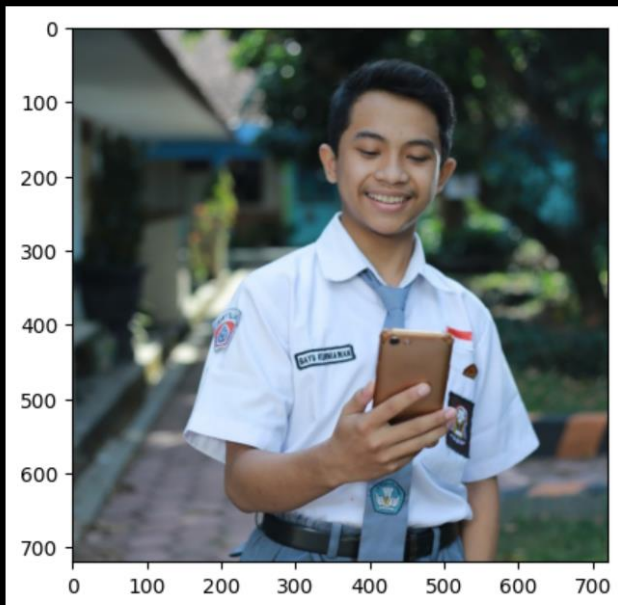
       [[ 50,  64,  67],
       [ 50,  64,  67],
       [ 50,  64,  67],
       ...,
       [ 23,  29,  29],
       [ 25,  31,  31],
       [ 26,  32,  32]],

       ...,

       [[ 66, 102,  64],
       [ 68, 104,  66],
       [ 70, 106,  68]]], dtype=uint8)
```

Analisis: Import beberapa library dan menampilkan array dari data citra yang digunakan

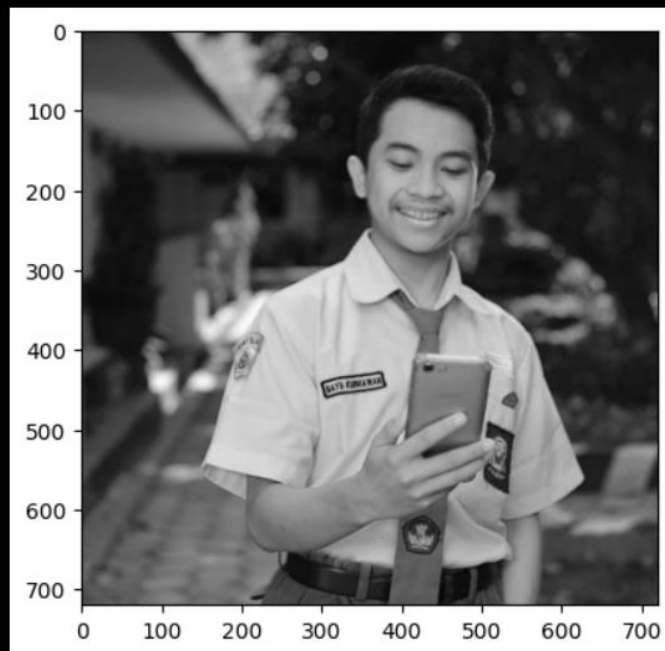
```
imgplot = plt.imshow(img)
```



Analisis: Menampilkan data citra yang digunakan

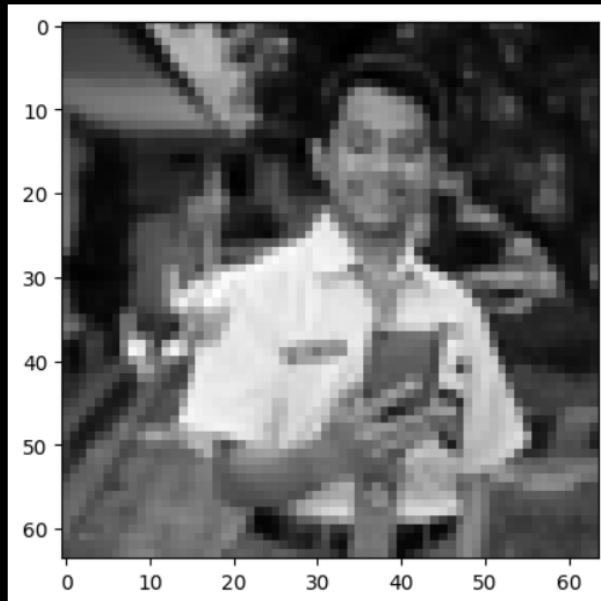
```
lum_img = img[:, :, 0]  
plt.imshow(lum_img)
```

<matplotlib.image.AxesImage at 0x1e18e80ded0>



Analisis: Kita bisa mengubah tampilan dari peta warna default

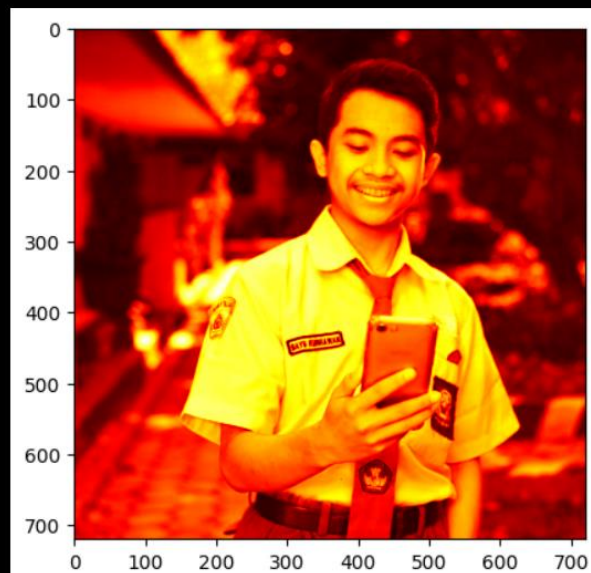
```
plt.imshow(img, cmap='gray')  
plt.show()
```



Analisis: Mengubah data citra menjadi grayscale (abu-abu)

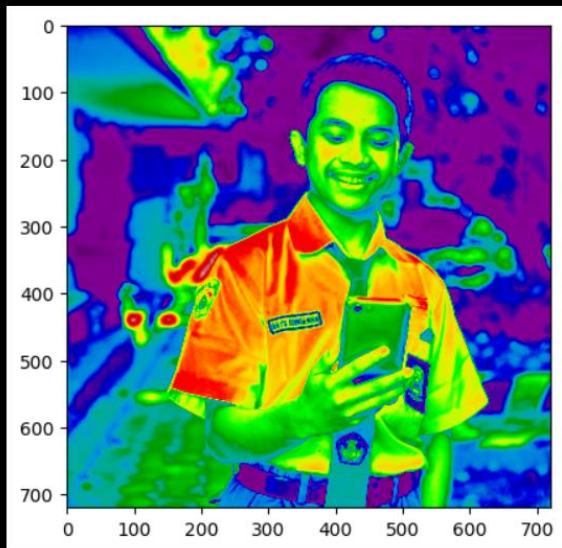
```
plt.imshow(lum_img, cmap="hot")
```

<matplotlib.image.AxesImage at 0x1e18e878df0>



Analisis : Mengubah peta warna pada objek plot yang ada menggunakan metode `set_cmap()`, peta warna dapat diubah dalam kondisi panas.

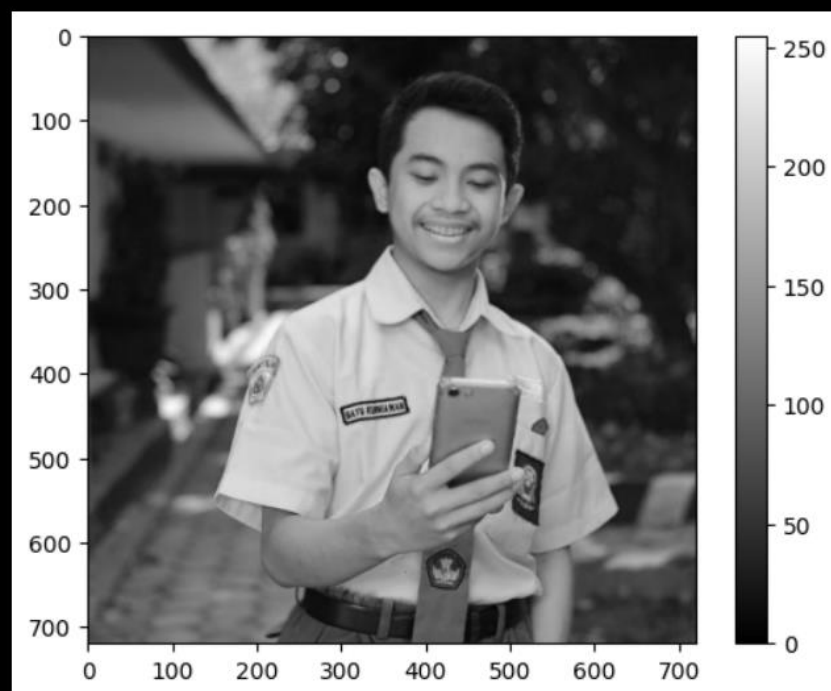
```
imgplot = plt.imshow(lum_img)
imgplot.set_cmap('nipy_spectral')
```



Analisis : Mengubah peta warna pada objek plot yang ada menggunakan metode `set_cmap()`, peta warna dapat diubah dalam kondisi .

```
imgplot = plt.imshow(lum_img)
plt.colorbar()
```

<matplotlib.colorbar.Colorbar at 0x1e18e9be320>

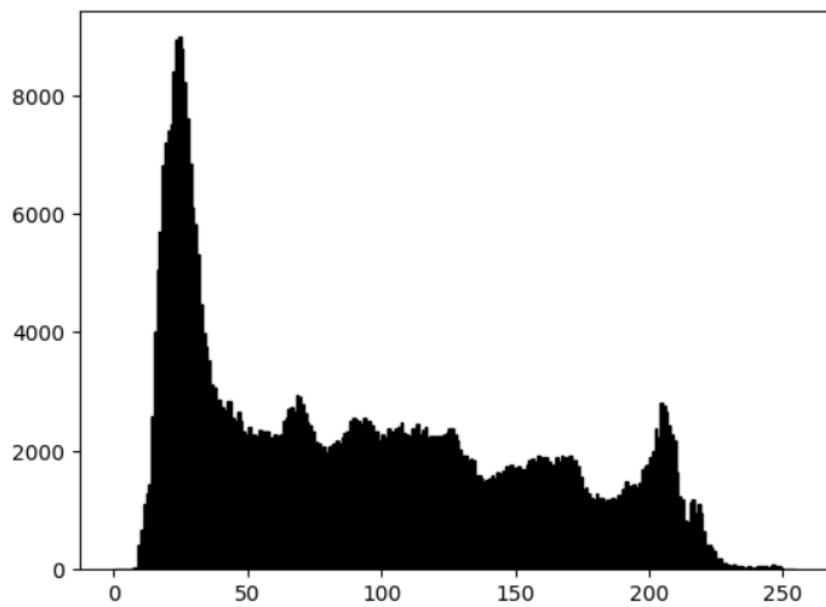


Analisis : Dapat untuk mengecek parameter warna hitam putih

```
plt.hist(lum_img.ravel(), bins=range(256), fc='k', ec='k')
```

Output exceeds the [size limit](#). Open the full output data [in a text editor](#)

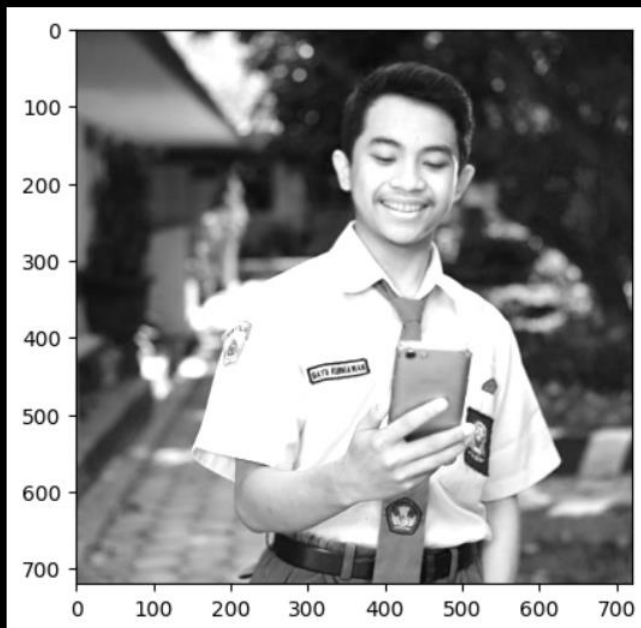
```
(array([1.000e+00, 0.000e+00, 3.000e+00, 3.000e+00, 5.000e+00, 1.300e+01,
        1.600e+01, 2.900e+01, 3.600e+01, 4.210e+02, 6.800e+02, 1.097e+03,
        1.275e+03, 1.426e+03, 2.590e+03, 4.018e+03, 5.071e+03, 5.691e+03,
        6.826e+03, 7.197e+03, 7.408e+03, 7.506e+03, 8.411e+03, 8.937e+03,
        8.980e+03, 8.788e+03, 8.219e+03, 7.610e+03, 6.837e+03, 6.092e+03,
        5.823e+03, 5.322e+03, 4.483e+03, 3.975e+03, 3.753e+03, 3.521e+03,
        3.110e+03, 3.069e+03, 2.835e+03, 2.869e+03, 2.735e+03, 2.694e+03,
        2.841e+03, 2.847e+03, 2.570e+03, 2.513e+03, 2.655e+03, 2.499e+03,
        2.328e+03, 2.268e+03, 2.402e+03, 2.197e+03, 2.288e+03, 2.255e+03,
        2.345e+03, 2.332e+03, 2.317e+03, 2.316e+03, 2.216e+03, 2.121e+03,
        2.269e+03, 2.250e+03, 2.278e+03, 2.499e+03, 2.570e+03, 2.707e+03,
        2.744e+03, 2.648e+03, 2.936e+03, 2.926e+03, 2.795e+03, 2.625e+03,
        2.476e+03, 2.423e+03, 2.324e+03, 2.141e+03, 2.120e+03, 2.002e+03,
        2.070e+03, 1.981e+03, 2.061e+03, 2.111e+03, 2.127e+03, 2.164e+03,
        2.129e+03, 2.215e+03, 2.304e+03, 2.341e+03, 2.507e+03, 2.564e+03,
        2.542e+03, 2.498e+03, 2.462e+03, 2.563e+03, 2.386e+03, 2.511e+03,
        2.422e+03, 2.319e+03, 2.337e+03, 2.168e+03, 2.275e+03, 2.225e+03,
        2.367e+03, 2.273e+03, 2.341e+03, 2.385e+03, 2.404e+03, 2.480e+03,
        2.302e+03, 2.279e+03, 2.260e+03, 2.368e+03, 2.379e+03, 2.453e+03,
        2.289e+03, 2.348e+03, 2.395e+03, 2.230e+03, 2.252e+03, 2.249e+03,
        2.223e+03, 2.260e+03, 2.248e+03, 2.274e+03, 2.306e+03, 2.374e+03,
        2.384e+03, 2.271e+03, 2.163e+03, 2.010e+03, 1.926e+03, 1.929e+03,
        1.837e+03, 1.866e+03, 1.853e+03, 1.581e+03, 1.578e+03, 1.548e+03,
        1.480e+03, 1.515e+03, 1.538e+03, 1.600e+03, 1.532e+03, 1.640e+03,
        1.608e+03, 1.669e+03, 1.736e+03, 1.730e+03, 1.765e+03, 1.723e+03,
        ...
        220., 221., 222., 223., 224., 225., 226., 227., 228., 229., 230.,
        231., 232., 233., 234., 235., 236., 237., 238., 239., 240., 241.,
        242., 243., 244., 245., 246., 247., 248., 249., 250., 251., 252.,
        253., 254., 255.]),
<BarContainer object of 255 artists>)
```



Analisis : Melihat parameter peta warna dalam bentuk histogram

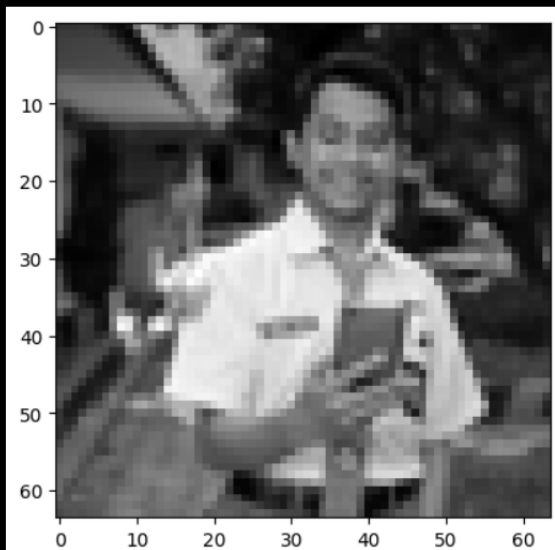
```
plt.imshow(lum_img, clim=(0, 175))
```

<matplotlib.image.AxesImage at 0x1e18e83e0b0>



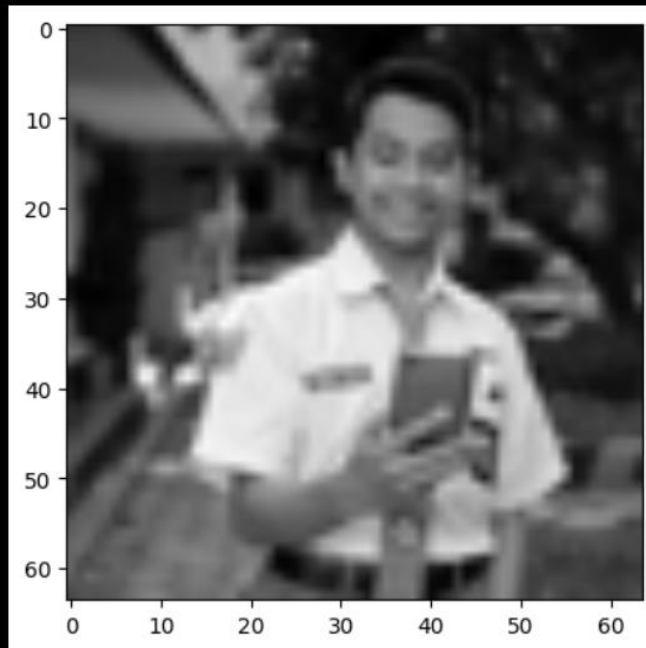
Analisis : Menentukan rentang dalam parameter warna, biasanya warna hitam putih memiliki rentang 0 -250, kita bisa mengubahnya seperti diubah menjadi 0 – 175.

```
img = Image.open("C:/Users/bayuk/OneDrive/Documents/AI/pens/smtr2/Pemrosesan Data/3/bayu1.jpg")
img.thumbnail((64, 64)) # resizes image in-place
imgplot = plt.imshow(img)
```



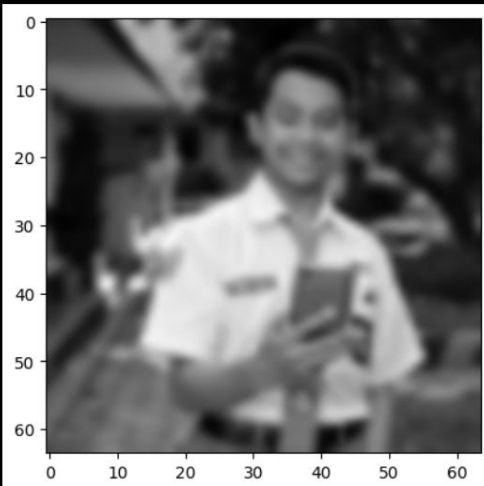
Analisis : Kita juga bisa mengubah ukuran gambar, semula gambar ukuran 700 x 700 kita bisa mengubah ukuran gambar tersebut seperti diubah pada ukuran 64x64, agar menghemat dalam penyimpanan gambar.


```
imgplot = plt.imshow(img, interpolation="bilinear")
```



Analisis : Kita bisa mengubah gambar menjadi bilinear untuk digunakan secara ekstensif dalam melakukan zooming, shrinking, rotating, dan geometric corrections

```
imgplot = plt.imshow(img, interpolation="bicubic")
```



Analisis : Kita juga dapat menggunakan bicubic interpolasi cubic untuk interpolasi data dalam bentuk dua dimensi