

KELOMPOK 4

1. Aas Novitasari (312210167)
2. Fathia Wardah S.Djawas (312210196)
3. Syifa Aurellia Rahma (312210009)
4. Tiara Putri (312210064)

MATA KULIAH : PENGOLAHAN CITRA

TUGAS :

Buatkan project untuk menyelesaikan segemntasi gambar citra digital dengan Algorithm KNN dengan kentuan berikut ini :

- Pilih jumlah cluster yang ingin dicari yaitu k.
- Tetapkan titik data secara acak ke salah satu k cluster.
- Kemudian hitung pusat clusternya.
- Hitung jarak titik data dari pusat masing-masing cluster.
- Bergantung pada jarak setiap titik data dari klaster, tetapkan kembali titik data ke klaster terdekat.
- Hitung lagi pusat cluster yang baru.
- Ulangi langkah 4,5 dan 6 hingga titik data tidak mengubah cluster, atau hingga kami mencapai jumlah iterasi yang ditetapkan.

Source code :

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

# Menampilkan plot dalam notebook
%matplotlib inline

# Membaca gambar (gunakan path yang sesuai dengan lokasi gambar
Anda)
```

```
image = cv2.imread("C:/Users/SYIFA
AURELLIA/Downloads/flower.jpg")

# Mengubah warna gambar dari BGR ke RGB
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)

# Menampilkan gambar asli
plt.imshow(image)
plt.title('Original Image')
plt.show()

# Membentuk ulang gambar menjadi susunan piksel 2D dengan 3
nilai warna (RGB)
pixel_vals = image.reshape((-1, 3))

# Mengkonversikan ke tipe float
pixel_vals = np.float32(pixel_vals)

# Menentukan kriteria agar algoritme berhenti berjalan: 100
iterasi atau epsilon (akurasi yang dibutuhkan) mencapai 85%
criteria = (cv2.TERM_CRITERIA_EPS + cv2.TERM_CRITERIA_MAX_ITER,
100, 0.85)

# Melakukan k-means clustering dengan jumlah cluster yang
ditetapkan sebagai 3 dan pusat acak pada awalnya dipilih
k = 3

retval, labels, centers = cv2.kmeans(pixel_vals, k, None,
criteria, 10, cv2.KMEANS_RANDOM_CENTERS)

# Mengonversi data pusat cluster menjadi nilai 8-bit
centers = np.uint8(centers)
```

```
# Menggunakan label untuk membentuk ulang data menjadi dimensi
gambar asli

segmented_data = centers[labels.flatten()]

segmented_image = segmented_data.reshape((image.shape))

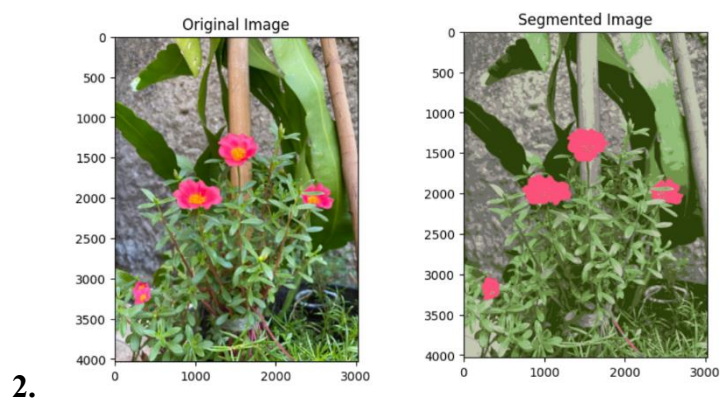
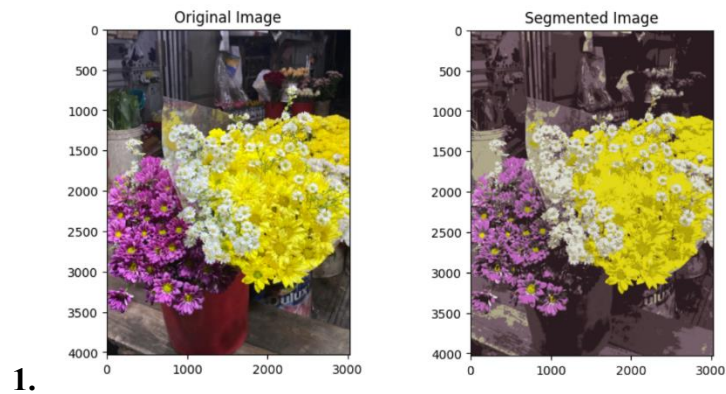
# Menampilkan gambar tersegmentasi

plt.imshow(segmented_image)

plt.title('Segmented Image')

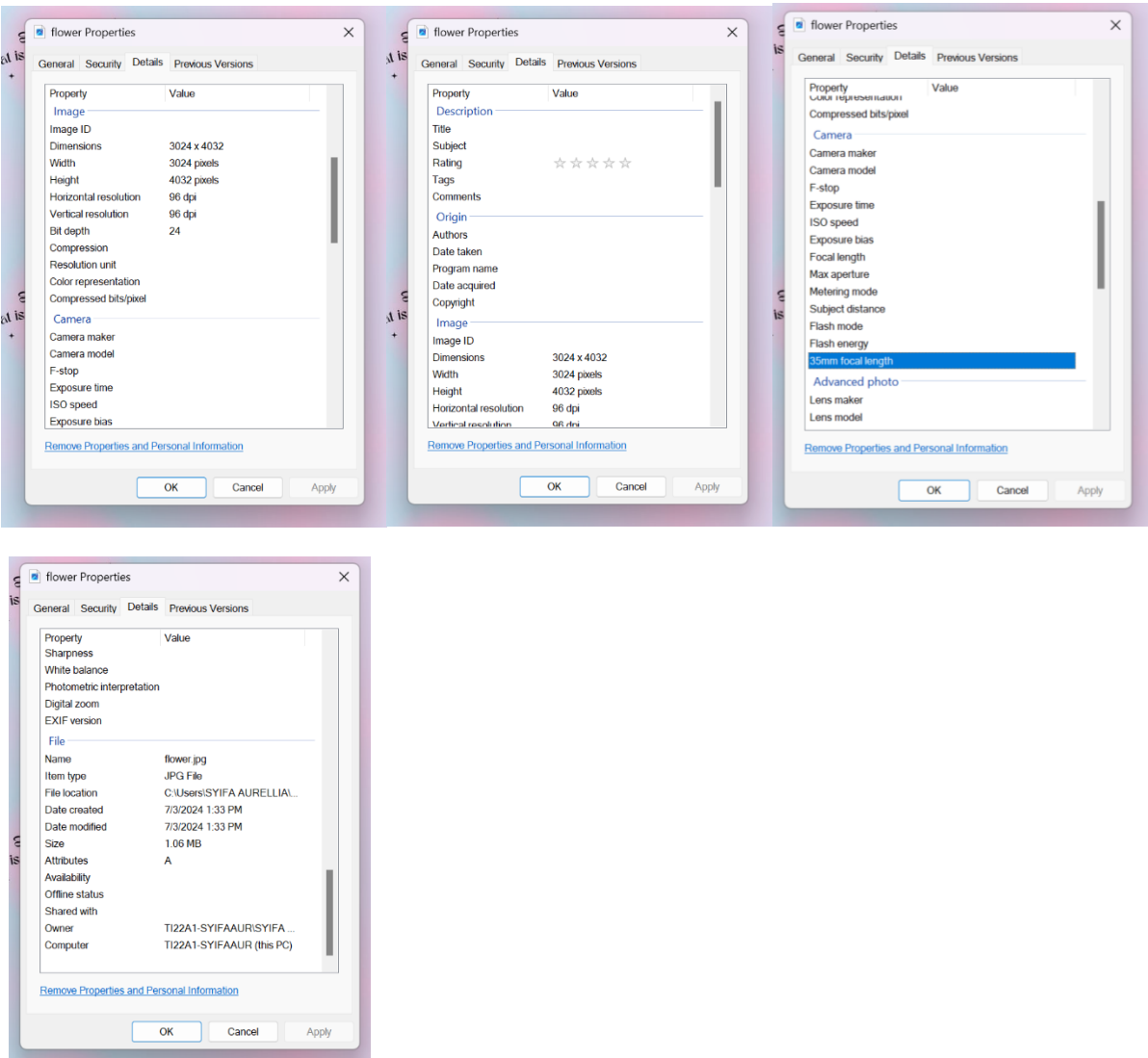
plt.show()
```

HASIL RUN :



Meta Data :

1.



2.

