

Attala Muflih Gumilang

1227030007

Praktikum Fisika Komputasi

A. Buatlah database dengan target warna hijau dan hitam!

B	G	R	Target
37	75	189	merah
45	75	184	merah
45	74	188	merah
48	75	186	merah
43	74	189	merah
46	78	197	merah
42	76	194	merah
36	75	192	merah
35	75	194	merah
36	75	193	merah
38	76	191	merah
42	75	191	merah
48	75	190	merah
49	75	190	merah
48	77	189	merah
49	76	193	merah
48	76	193	merah
46	75	191	merah
44	77	190	merah
43	77	188	merah
39	77	191	merah
37	76	190	merah
39	75	192	merah
44	74	193	merah
42	75	191	merah
46	83	201	merah
46	83	202	merah
42	84	200	merah
42	84	199	merah
42	84	199	merah
41	84	199	merah
43	84	200	merah
42	84	199	merah
42	84	198	merah
44	84	200	merah
49	75	184	merah
49	74	188	merah

47,73,185,merah
42,74,185,merah
38,74,185,merah
36,74,185,merah
36,73,184,merah
33,73,179,merah
32,70,178,merah
34,69,180,merah
36,67,178,merah
37,68,179,merah
40,67,179,merah
41,69,175,merah
39,70,172,merah
34,70,174,merah
36,68,172,merah
38,69,170,merah
40,68,172,merah
40,68,171,merah
36,67,177,merah
36,66,175,merah
39,65,177,merah
39,66,174,merah
38,67,171,merah
40,68,170,merah
40,68,170,merah
43,68,170,merah
43,68,170,merah
44,68,170,merah
29,69,172,merah
33,69,171,merah
33,68,174,merah
36,68,177,merah
38,66,178,merah
35,31,25,hitam
32,29,26,hitam
28,28,26,hitam
27,28,22,hitam
30,28,24,hitam
34,29,22,hitam
32,27,28,hitam
29,26,26,hitam
30,26,27,hitam
45,36,30,hitam
46,36,32,hitam
47,36,31,hitam
44,36,31,hitam
37,34,31,hitam
34,33,29,hitam
33,33,28,hitam
34,35,29,hitam

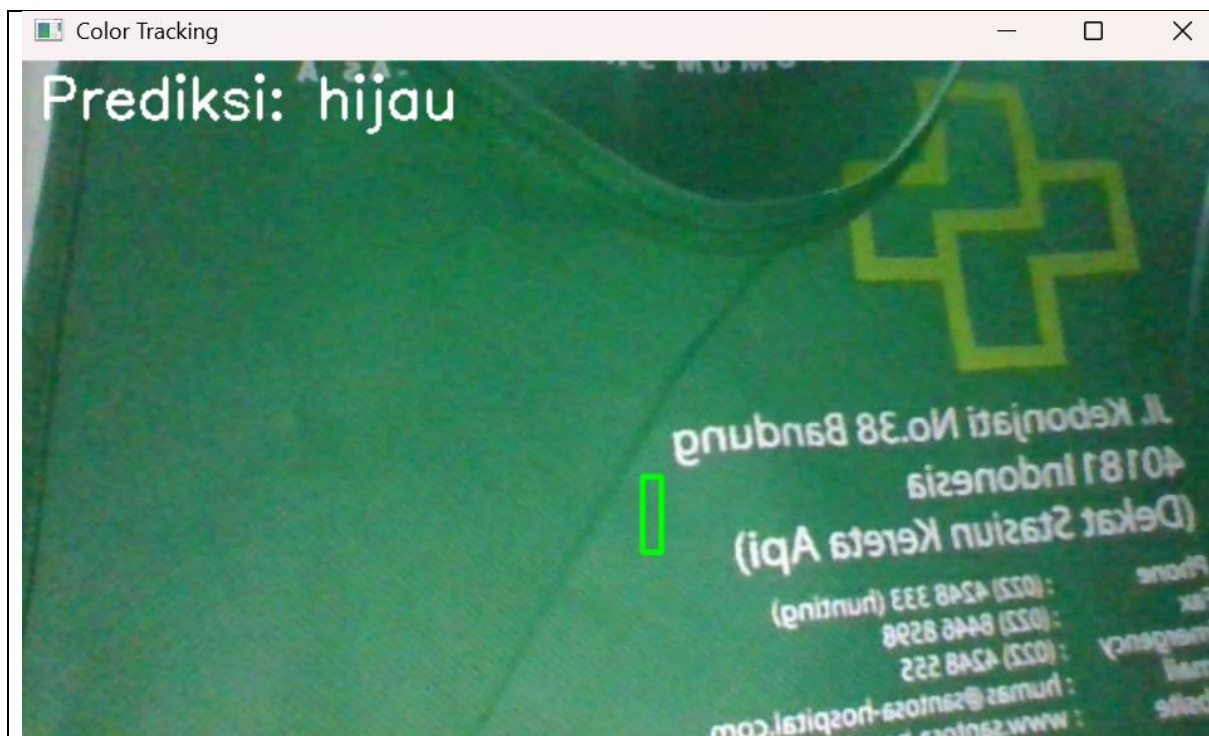
34,35,29,hitam
37,33,30,hitam
41,37,30,hitam
60,145,136,kuning
61,146,133,kuning
63,146,131,kuning
63,146,133,kuning
61,145,134,kuning
62,146,133,kuning
65,146,133,kuning
65,146,133,kuning
63,145,131,kuning
63,144,132,kuning
65,145,139,kuning
67,145,138,kuning
66,145,138,kuning
65,145,138,kuning
65,145,138,kuning
65,146,138,kuning
67,146,135,kuning
67,145,135,kuning
69,143,134,kuning
70,143,134,kuning
62,34,26,biru tua
63,34,25,biru tua
62,34,27,biru tua
61,35,26,biru tua
61,34,26,biru tua
62,34,26,biru tua
60,33,25,biru tua
62,32,22,biru tua
60,32,23,biru tua
65,33,28,biru tua
65,34,27,biru tua
62,35,25,biru tua
64,34,26,biru tua
60,32,24,biru tua
63,34,27,biru tua
64,34,27,biru tua
62,34,27,biru tua
63,34,27,biru tua
63,34,27,biru tua
65,35,28,biru tua
146,84,27,biru muda
145,84,26,biru muda
145,85,26,biru muda
143,85,26,biru muda
144,85,24,biru muda
143,87,33,biru muda
141,86,34,biru muda

143,86,33,biru muda
142,85,32,biru muda
143,84,33,biru muda
144,85,30,biru muda
142,84,33,biru muda
142,84,30,biru muda
140,84,31,biru muda
141,83,28,biru muda
144,88,33,biru muda
144,88,32,biru muda
145,87,34,biru muda
143,87,33,biru muda

119,164,67,hijau
106,154,69,hijau
110,159,69,hijau
109,152,71,hijau
106,154,74,hijau
101,145,69,hijau
99,142,72,hijau
98,140,63,hijau
103,148,68,hijau
101,146,68,hijau
98,145,70,hijau
100,142,69,hijau
87,124,50,hijau
85,122,52,hijau
86,124,51,hijau
82,123,49,hijau
90,126,59,hijau
90,124,57,hijau
100,141,67,hijau
105,148,66,hijau
146,128,139,hitam
141,122,136,hitam
139,120,137,hitam
155,142,152,hitam
159,144,155,hitam
1,2,1,hitam
1,2,2,hitam
1,2,1,hitam
1,2,2,hitam
1,2,2,hitam
1,2,1,hitam
2,2,1,hitam
1,3,1,hitam
3,5,3,hitam
7,10,7,hitam
97,91,91,hitam
110,105,106,hitam

177,171,173,hitam
215,209,212,hitam
218,209,213,hitam
207,193,200,hitam
208,195,199,hitam
190,172,178,hitam
176,151,162,hitam
145,126,132,hitam
86,87,105,hitam
180,160,169,hitam
185,163,173,hitam
182,161,168,hitam
19,19,18,hitam
14,16,17,hitam
18,18,19,hitam
23,25,26,hitam
17,18,18,hitam

B. Prediksi warna hijau dan hitam!





C. Jelaskan kode program dan prosesnya hingga diperoleh hasil prediksi dengan Bahasa sendiri siingkat, padat dan jelas!

Program pertama, *Ambil data.py*, digunakan untuk mengumpulkan data warna dari kamera dan menyimpannya ke dalam sebuah file database. Program ini membantu membuat sebuah koleksi data berupa nilai-nilai intensitas warna merah (R), hijau (G), dan biru (B), yang kemudian diberi label sesuai dengan nama warna tertentu, seperti merah, hijau, biru, dan sebagainya. Data ini kemudian digunakan oleh program kedua, *Baca semua.py*, yang memanfaatkan file database tadi untuk melatih model *machine learning* (dalam hal ini, SVM) sehingga dapat mengenali warna secara otomatis berdasarkan data yang dilihat oleh kamera.

Dengan program ini, pengguna dapat memeriksa apakah kamera terhubung dan berfungsi dengan baik sebelum digunakan dalam aplikasi lain, seperti program pengolahan citra atau pengenalan objek. Program ini cocok digunakan sebagai langkah awal untuk menguji koneksi perangkat keras kamera dengan komputer.

Kodingan ambil data

```
import cv2
import numpy as np
import csv

# Konfigurasi Kamera
cap = cv2.VideoCapture(0)
cap.set(cv2.CAP_PROP_FRAME_WIDTH, 480)
cap.set(cv2.CAP_PROP_FRAME_HEIGHT, 360)

# Nama file database
FileDB = "DatabaseWarna.txt"

# Header untuk file CSV
header = ['B', 'G', 'R', 'Target']

# Buat file CSV jika belum ada
try:
    with open(FileDB, 'x', newline='') as f:
        writer = csv.writer(f)
        writer.writerow(header)
except FileExistsError:
    print(f'{FileDB} sudah ada, melanjutkan penambahan data.')

print("Tekan tombol berikut untuk menambahkan data warna:")
print("1: Merah, 2: Hijau, 3: Biru, 4: Hitam, 5: Kuning, 6: Putih, ESC: Keluar")

while True:
    ret, img = cap.read()
    if not ret:
        print("Gagal membaca frame dari kamera.")
        break

# Membalikkan kamera jika terbalik
```

```

img = cv2.flip(img, 1)

# Ambil warna rata-rata dari area tertentu
region = img[220:260, 330:340] # Area yang dianalisis
colorB = int(np.mean(region[:, :, 0]))
colorG = int(np.mean(region[:, :, 1]))
colorR = int(np.mean(region[:, :, 2]))
color = [colorB, colorG, colorR]

# Tampilkan area analisis dan warna rata-rata
cv2.rectangle(img, (330, 220), (340, 260), (0, 255, 0), 2)
cv2.putText(
    img,
    f'B: {colorB}, G: {colorG}, R: {colorR}',
    (10, 30),
    cv2.FONT_HERSHEY_SIMPLEX,
    1,
    (255, 255, 255),
    2,
)

cv2.imshow("Database Color Capture", img)

# Deteksi tombol untuk menentukan warna
key = cv2.waitKey(30) & 0xFF
if key == ord('1'): # Merah
    label = "merah"
elif key == ord('2'): # Hijau
    label = "hijau"
elif key == ord('3'): # Biru
    label = "biru"
elif key == ord('4'): # Hitam
    label = "hitam"
elif key == ord('5'): # Kuning
    label = "kuning"
elif key == ord('6'): # Putih
    label = "putih"
elif key == 27: # ESC untuk keluar
    break
else:
    continue

# Simpan data ke file CSV
with open(FileDB, 'a', newline='') as f:
    writer = csv.writer(f)
    writer.writerow(color + [label])
    print(f'Data {color} dengan label '{label}' telah disimpan.')

cap.release()
cv2.destroyAllWindows()

```


Kodingan cek kamera

```
import cv2
import numpy as np

# Membuka kamera
cap = cv2.VideoCapture(0)

while True:
    # Membaca frame dari kamera
    ret, frame = cap.read()

    if not ret:
        print("Gagal membaca frame dari kamera.")
        break

    # Membalik frame secara horizontal
    frame = cv2.flip(frame, 1)

    # Menampilkan frame
    cv2.imshow("Camera", frame)

    # Menunggu tombol tekan, keluar jika tombol ESC ditekan
    key = cv2.waitKey(1)
    if key == 27: # Tombol ESC
        break

# Melepaskan kamera dan menutup jendela
cap.release()
cv2.destroyAllWindows()
```

Kodingan menjalankan program

```
import cv2
import numpy as np
import csv
import time
from sklearn import svm
import pandas as pd
from sklearn.preprocessing import StandardScaler

# Konfigurasi Kamera
cap = cv2.VideoCapture(0)
cap.set(cv2.CAP_PROP_FRAME_WIDTH, 480)
cap.set(cv2.CAP_PROP_FRAME_HEIGHT, 360)

# Membaca Database
FileDB = "Databasewarna.txt" # Pastikan file ini tersedia dan formatnya benar

try:
    Database = pd.read_csv(FileDB, sep=",", header=0)
    print("Database: \n", Database)
```

```

except FileNotFoundError:
    print(f"File {FileDB} tidak ditemukan. Pastikan file tersedia.")
    exit()

# X Data (B, G, R), y Target
X = Database[['B', 'G', 'R']]
y = Database['Target']

# Normalisasi Data dan Pelatihan Model SVM
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X) # Normalisasi data
clf = svm.SVC(kernel='linear') # Gunakan kernel linear
clf.fit(X_scaled, y)

# Fungsi Prediksi Warna
def predict_color(b, g, r):
    color_scaled = scaler.transform([[b, g, r]])
    try:
        prediction = clf.predict(color_scaled)[0] # Ambil hasil prediksi
        return prediction
    except Exception as e:
        return "Tidak Teridentifikasi"

# Loop Kamera untuk Prediksi
while True:
    ret, img = cap.read()
    if not ret:
        print("Gagal membaca frame dari kamera.")
        break

    # Membalikkan kamera jika terbalik
    img = cv2.flip(img, 1)

    # Ambil warna rata-rata dari area tertentu
    region = img[220:260, 330:340] # Area yang dianalisis
    colorB = int(np.mean(region[:, :, 0]))
    colorG = int(np.mean(region[:, :, 1]))
    colorR = int(np.mean(region[:, :, 2]))
    color = [colorB, colorG, colorR]

    # Prediksi warna
    prediction = predict_color(colorB, colorG, colorR)
    print(f"B: {colorB}, G: {colorG}, R: {colorR}, Prediksi: {prediction}")

    # Tampilkan hasil di jendela kamera
    cv2.putText(
        img,
        f"Prediksi: {prediction}",
        (10, 30),
        cv2.FONT_HERSHEY_SIMPLEX,

```

```
1,  
(255, 255, 255),  
2,  
)  
cv2.rectangle(img, (330, 220), (340, 260), (0, 255, 0), 2) # Area analisis  
cv2.imshow("Color Tracking", img)  
  
# Tombol keluar (ESC)  
k = cv2.waitKey(30) & 0xFF  
if k == 27: # Tekan ESC untuk keluar  
    break  
  
cap.release()  
cv2.destroyAllWindows()
```