

Nama : Ulfah Hasanah

NIM : 1227030036

1. 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

11,144,59,hijau
1,137,50,hijau
6,139,58,hijau
6,141,57,hijau
2,141,62,hijau
0,142,63,hijau
5,141,63,hijau
0,141,63,hijau
2,140,58,hijau
2,141,62,hijau
2,141,60,hijau
1,141,61,hijau
1,141,62,hijau
1,141,60,hijau
6,149,78,hijau
6,151,76,hijau
6,152,76,hijau
5,150,77,hijau
5,145,67,hijau
3,144,62,hijau
3,143,67,hijau
1,141,64,hijau
3,141,62,hijau
4,142,61,hijau
3,141,65,hijau
1,141,63,hijau
1,142,66,hijau
6,145,68,hijau
5,145,67,hijau
3,145,70,hijau
4,146,71,hijau
5,147,73,hijau
7,147,78,hijau
2,144,67,hijau
4,146,69,hijau
5,146,70,hijau
3,143,68,hijau
2,143,66,hijau
3,141,62,hijau
9,142,66,hijau

9,143,69,hijau
5,145,71,hijau
6,141,68,hijau
4,142,70,hijau
5,146,70,hijau
11,151,79,hijau
8,149,76,hijau
6,148,75,hijau
2,146,71,hijau
6,147,73,hijau
3,147,67,hijau
5,150,73,hijau
8,151,75,hijau
9,151,79,hijau
6,152,76,hijau
10,154,81,hijau
14,155,81,hijau
22,158,88,hijau
3,148,67,hijau
8,149,72,hijau
4,146,66,hijau
8,150,70,hijau
6,149,69,hijau
1,147,65,hijau
4,147,66,hijau
12,149,69,hijau
4,148,70,hijau
10,148,69,hijau
10,148,68,hijau
7,148,67,hijau
7,148,68,hijau
6,148,68,hijau
6,148,72,hijau
4,147,69,hijau
5,147,65,hijau
9,148,67,hijau
7,149,72,hijau
6,150,74,hijau
6,147,67,hijau
6,148,66,hijau
8,148,71,hijau
8,147,69,hijau
7,148,68,hijau
8,147,66,hijau
6,148,71,hijau
8,151,72,hijau
7,148,69,hijau
8,150,71,hijau
3,150,71,hijau
11,150,75,hijau

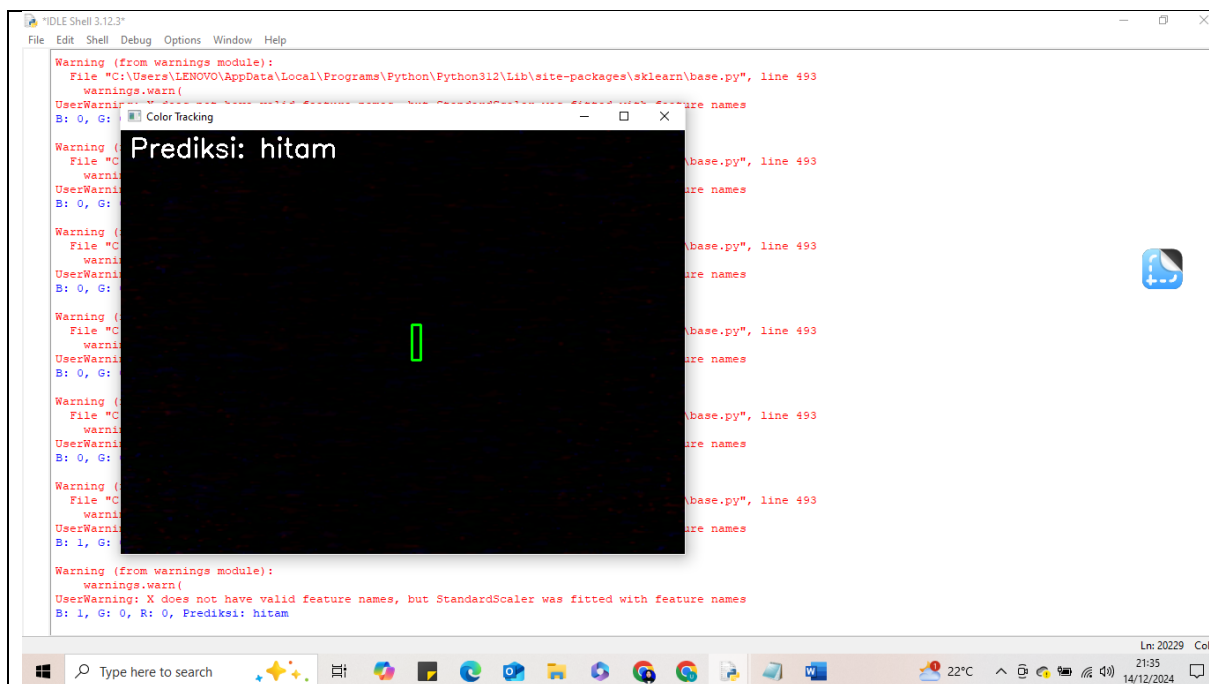
6,150,71,hijau
12,146,67,hijau
9,145,65,hijau
7,146,68,hijau
8,148,67,hijau
4,146,66,hijau
12,145,69,hijau
5,143,68,hijau
3,144,62,hijau
7,144,63,hijau
5,145,68,hijau
4,146,65,hijau
9,144,65,hijau
11,144,65,hijau
2,144,64,hijau
5,145,64,hijau
2,144,62,hijau
2,144,63,hijau
2,143,60,hijau
6,145,67,hijau
8,145,65,hijau
5,145,63,hijau
3,145,64,hijau
1,145,65,hijau
3,144,68,hijau
5,144,66,hijau
4,144,66,hijau
7,144,68,hijau
0,0,0,hitam
0,0,1,hitam
0,0,0,hitam
0,0,1,hitam
0,0,0,hitam
0,0,1,hitam
0,0,0,hitam
0,0,1,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,1,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,1,hitam
0,0,0,hitam
0,0,0,hitam
0,0,1,hitam
0,0,0,hitam
0,0,1,hitam

0,0,1,hitam
0,0,1,hitam
0,0,2,hitam
0,0,1,hitam
0,0,1,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
1,0,0,hitam
0,0,0,hitam
0,0,0,hitam
1,0,1,hitam
1,0,0,hitam
0,0,1,hitam
0,0,0,hitam
0,0,0,hitam
0,0,1,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,1,hitam
0,0,1,hitam
0,0,0,hitam
0,0,0,hitam
1,0,2,hitam
0,0,1,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,1,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,1,hitam
1,0,0,hitam
0,0,2,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam

0,0,1,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,2,hitam
0,0,1,hitam
0,0,1,hitam
0,0,1,hitam
0,0,1,hitam
0,0,1,hitam
0,0,0,hitam
0,0,0,hitam
1,0,0,hitam
1,0,1,hitam
0,0,0,hitam
0,0,2,hitam
0,0,1,hitam
0,0,0,hitam
1,0,0,hitam
0,0,0,hitam
0,0,1,hitam
0,0,1,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,1,hitam
0,0,0,hitam
0,0,1,hitam
0,0,1,hitam
0,0,1,hitam
0,0,1,hitam
0,0,0,hitam
0,0,1,hitam
0,0,1,hitam
0,0,4,hitam
0,0,1,hitam
0,0,1,hitam
0,0,0,hitam
1,0,1,hitam
0,1,0,hitam
0,2,0,hitam
0,2,0,hitam
1,0,0,hitam
0,0,0,hitam
1,0,2,hitam
0,0,0,hitam
0,0,2,hitam
0,0,1,hitam

1,0,1,hitam
1,0,2,hitam
1,0,0,hitam
0,0,0,hitam
0,0,1,hitam
0,0,1,hitam
0,0,0,hitam
1,0,0,hitam
0,0,1,hitam
0,0,0,hitam
1,0,0,hitam
0,0,0,hitam
0,1,0,hitam
3,0,0,hitam
0,0,0,hitam
1,0,0,hitam
0,0,0,hitam
1,0,0,hitam
0,1,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
0,0,0,hitam
1,0,0,hitam
1,0,0,hitam
0,0,0,hitam
1,0,0,hitam
0,0,0,hitam
1,0,1,hitam
0,0,1,hitam
0,0,1,hitam
0,0,1,hitam
0,0,1,hitam
0,0,1,hitam
2,0,1,hitam
0,0,1,hitam
3,0,1,hitam
1,0,1,hitam
0,0,1,hitam
1,0,2,hitam

2. Prediksi warna hijau dan hitam!



3. Jelaskan kode program dan prosesnya hingga diperoleh hasil prediksi !

Kodingan cek kamera

```
import cv2
import numpy as np
cap = cv2.VideoCapture(0)
```

```
while True:
    ret, frame = cap.read()
    frame = cv2.flip (frame, 1)
    cv2.imshow ("camera", frame)
    key = cv2.waitKey (1)
    if key == 27:
        break
cap.release ()
cv2.destroyAllWindows ()
```

Penjelasan : Program ini memulai dengan mengaktifkan kamera utama komputer, yang secara default teridentifikasi sebagai kamera dengan ID 0. Setelah kamera aktif, program membaca setiap frame video yang dihasilkan oleh kamera dan menampilkannya di jendela aplikasi. Untuk memberikan tampilan seperti cermin, setiap frame yang ditampilkan dibalik secara horizontal menggunakan fungsi cv2.flip. Program ini terus berjalan sambil menunggu kita sebagai pengguna menekan tombol ESC (kode 27) pada keyboard, yang berfungsi untuk menghentikan dan menutup aplikasi.

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()

```

Penjelasan : Program ini mengatur kamera dengan resolusi 480x360 untuk membaca video, lalu fokus pada sebuah area kecil di layar (koordinat 220:260, 330:340) untuk menganalisis warna. Rata-rata nilai intensitas warna biru (B), hijau (G), dan merah (R) dari area tersebut dihitung menggunakan metode `np.mean()`, yang menghasilkan nilai rata-rata untuk setiap komponen warna berdasarkan piksel di area yang dianalisis. Kita dapat memberikan label warna seperti merah, hijau, biru, hitam, kuning, atau putih dengan menekan tombol angka. Data warna yang sudah dilabeli ini kemudian disimpan dalam file `DatabaseWarna.txt` dalam format `[B, G, R, Target]`.

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()
```

Penjelasannya: Program ini membaca data warna dari file *Databasewarna.txt*, menggunakan kolom nilai biru (B), hijau (G), dan merah (R) sebagai fitur (X) dan kolom *Target* sebagai label (y). Data tersebut dinormalisasi dengan *StandardScaler* untuk menyamakan skala, lalu dilatih menggunakan model SVM dengan kernel linear. Setelah model siap, program membaca nilai rata-rata B, G, dan R dari area tertentu pada kamera, menormalisasinya, dan memprediksi warna menggunakan model SVM. Hasil prediksi warna ditampilkan langsung di layar video dengan bantuan fungsi *cv2.putText*

Dengan demikian semua proses/alur algoritma kode program keseluruhan diatas, singkatnya dimulai dengan program membaca data dari kamera dan menampilkan video langsung di layar. Dalam proses ini, intensitas warna dari area tertentu pada video dihitung dengan mendapatkan nilai rata-rata warna biru (B), hijau (G), dan merah (R) menggunakan *np.mean()*. Kita kemudian dapat memberikan label pada warna yang dianalisis berdasarkan intensitas tersebut, dan data warna yang sudah dilabeli disimpan dalam file database. File ini kemudian digunakan untuk melatih model pembelajaran mesin menggunakan algoritma SVM, yang dirancang untuk mengenali pola warna berdasarkan data yang telah dilabeli. Setelah model selesai dilatih, program menggunakan model tersebut untuk memprediksi warna dari input kamera secara real-time, dengan hasil prediksi yang langsung ditampilkan di layar video.