

Adinda Salsabila

1227030003

Modul 12

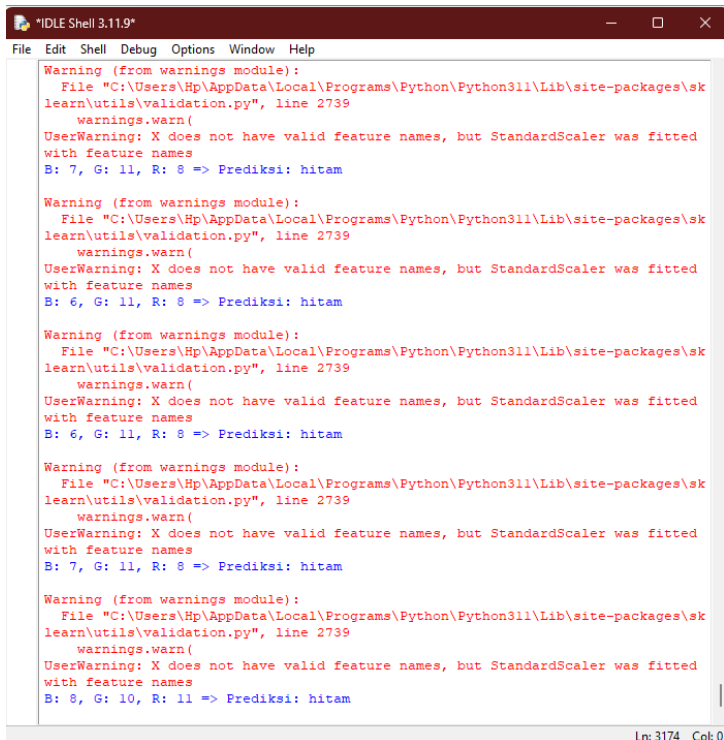
1. Database target warna hijau dan hitam



```
DatabaseWarna.txt
File Edit View

B,G,R,Target
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
85,137,86,hijau
55,143,103,hijau
```

2. Prediksi warna hitam



```
IDLE Shell 3.11.9*
File Edit Shell Debug Options Window Help

Warning (from warnings module):
  File "C:\Users\Hp\AppData\Local\Programs\Python\Python311\Lib\site-packages\sk
learn\utils\validation.py", line 2739
    warnings.warn(
UserWarning: X does not have valid feature names, but StandardScaler was fitted
with feature names
B: 7, G: 11, R: 8 => Prediksi: hitam

Warning (from warnings module):
  File "C:\Users\Hp\AppData\Local\Programs\Python\Python311\Lib\site-packages\sk
learn\utils\validation.py", line 2739
    warnings.warn(
UserWarning: X does not have valid feature names, but StandardScaler was fitted
with feature names
B: 6, G: 11, R: 8 => Prediksi: hitam

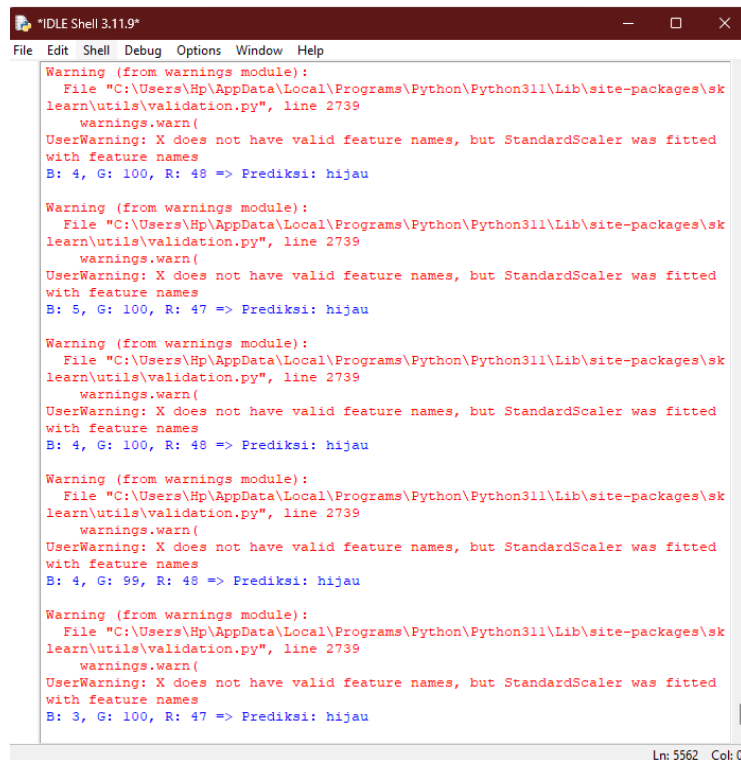
Warning (from warnings module):
  File "C:\Users\Hp\AppData\Local\Programs\Python\Python311\Lib\site-packages\sk
learn\utils\validation.py", line 2739
    warnings.warn(
UserWarning: X does not have valid feature names, but StandardScaler was fitted
with feature names
B: 6, G: 11, R: 8 => Prediksi: hitam

Warning (from warnings module):
  File "C:\Users\Hp\AppData\Local\Programs\Python\Python311\Lib\site-packages\sk
learn\utils\validation.py", line 2739
    warnings.warn(
UserWarning: X does not have valid feature names, but StandardScaler was fitted
with feature names
B: 7, G: 11, R: 8 => Prediksi: hitam

Warning (from warnings module):
  File "C:\Users\Hp\AppData\Local\Programs\Python\Python311\Lib\site-packages\sk
learn\utils\validation.py", line 2739
    warnings.warn(
UserWarning: X does not have valid feature names, but StandardScaler was fitted
with feature names
B: 8, G: 10, R: 11 => Prediksi: hitam

Ln: 3174 Col: 0
```

Prediksi warna hijau



```
*IDLE Shell 3.11.9*
File Edit Shell Debug Options Window Help

Warning (from warnings module):
  File "C:\Users\Hp\AppData\Local\Programs\Python\Python311\Lib\site-packages\sk
learn\utils\validation.py", line 2739
    warnings.warn(
UserWarning: X does not have valid feature names, but StandardScaler was fitted
with feature names
B: 4, G: 100, R: 48 => Prediksi: hijau

Warning (from warnings module):
  File "C:\Users\Hp\AppData\Local\Programs\Python\Python311\Lib\site-packages\sk
learn\utils\validation.py", line 2739
    warnings.warn(
UserWarning: X does not have valid feature names, but StandardScaler was fitted
with feature names
B: 5, G: 100, R: 47 => Prediksi: hijau

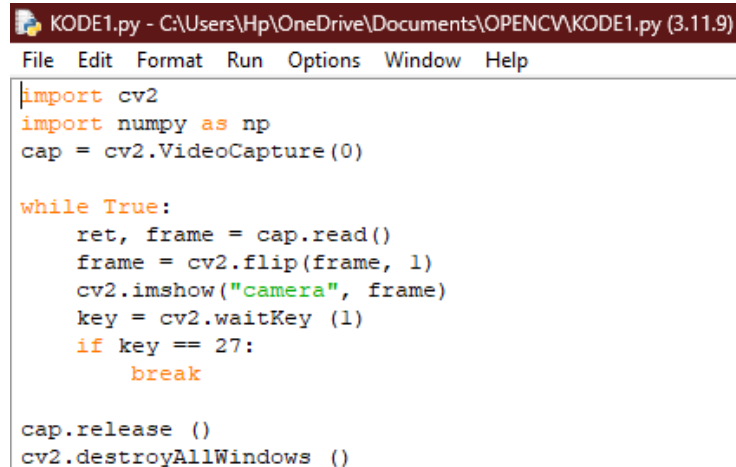
Warning (from warnings module):
  File "C:\Users\Hp\AppData\Local\Programs\Python\Python311\Lib\site-packages\sk
learn\utils\validation.py", line 2739
    warnings.warn(
UserWarning: X does not have valid feature names, but StandardScaler was fitted
with feature names
B: 4, G: 100, R: 48 => Prediksi: hijau

Warning (from warnings module):
  File "C:\Users\Hp\AppData\Local\Programs\Python\Python311\Lib\site-packages\sk
learn\utils\validation.py", line 2739
    warnings.warn(
UserWarning: X does not have valid feature names, but StandardScaler was fitted
with feature names
B: 4, G: 99, R: 48 => Prediksi: hijau

Warning (from warnings module):
  File "C:\Users\Hp\AppData\Local\Programs\Python\Python311\Lib\site-packages\sk
learn\utils\validation.py", line 2739
    warnings.warn(
UserWarning: X does not have valid feature names, but StandardScaler was fitted
with feature names
B: 3, G: 100, R: 47 => Prediksi: hijau

Ln: 5562 Col: 0
```

3. Penjelasan kode program



```
KODE1.py - C:\Users\Hp\OneDrive\Documents\OPENCV\KODE1.py (3.11.9)
File Edit Format Run Options Window Help

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()
```

Pada pemrograman kali ini menggunakan library cv2 yang berfungsi untuk pemrosesan gambar. Library numpy digunakan karena memakai software python karena gambar dan video direpresentasikan sebagai array.

Pada line 3, membuka kamera utama perangkat.

Pada line 6, membaca frame dari kamera.

Pada line 7, membalik frame secara horizontal agar terlihat seperti dicerminkan.

Pada line 8, menampilkan frame yang diperoleh dalam jendela bernama camera

Pada line 9, menunggu input setiap 1 ms.

Pada line 10, jika kode 27 ditekan maka loop akan berhenti.

Pada line 13, melepaskan akses kamera.

Pada line 14, menutup semua jendela OpenCV.

```
KODE2.py - C:\Users\Hp\OneDrive\Documents\OPENCV\KODE2.py (3.11.9)
File Edit Format Run Options Window Help

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

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

    # Ambil warna rata-rata dari area tertentu
    region = img[220:260, 300:340] # 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]
```

Kode tersebut adalah program Python yang menggunakan OpenCV untuk menangkap warna dari area tertentu pada kamera dan menyimpannya ke dalam file CSV sebagai database warna. Tujuan dari pemrograman ini yakni membuat database warna dengan nilai B,G,R dan label target yang dapat digunakan untuk pemrosesan lebih lanjut, seperti klasifikasi warna.

1. Konfigurasi Kamera

Program membuka kamera komputer dengan resolusi 480x360 piksel menggunakan `cv2.VideoCapture()`.

2. Inisialisasi File Database

File DatabaseWarna.txt dibuat dengan header kolom B, G, R, dan Target jika file belum ada. Jika file sudah ada, program akan menambahkan data baru tanpa membuat file baru.

3. Instruksi Pengguna

Program memberikan petunjuk kepada pengguna untuk menekan tombol tertentu (1-6) untuk memilih warna target, atau tombol ESC untuk keluar.

4. Pengolahan Warna

Setiap frame dari kamera diproses, dan area tertentu pada gambar (220:260, 300:340) dianalisis untuk menghitung nilai rata-rata warna biru (B), hijau (G), dan merah (R).

Nilai warna rata-rata ditampilkan di layar bersama dengan area analisis yang diberi kotak hijau.

5. Input Warna dan Penyimpanan Data

Ketika pengguna menekan tombol angka tertentu, program menetapkan label warna (merah, hijau, biru, dll.) sesuai tombol. Nilai rata-rata warna dan label tersebut disimpan dalam file CSV DatabaseWarna.txt.

6. Keluar Program

Pengguna dapat keluar dari program dengan menekan tombol ESC, setelah itu kamera dilepaskan dan semua jendela ditutup.

```
KODE3.py - C:\Users\Hp\OneDrive\Documents\OPENCV\KODE3.py (3.11.9)
File Edit Format Run Options Window Help

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

# 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)
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"
```

Kode tersebut adalah program Python untuk mendeteksi warna real-time menggunakan kamera dan memprediksi nama warna berdasarkan model SVM (Support Vector Machine) yang dilatih dengan data dari file database. Tujuan dari pemrograman ini yakni membaca warna dari kamera, memprediksi label warna berdasarkan model SVM yang telah dilatih, dan menampilkan hasil prediksi secara real-time. Program ini berguna untuk aplikasi seperti pelacakan warna atau pengenalan warna otomatis.

1. Konfigurasi Kamera

Kamera diinisialisasi dengan resolusi 480x360 piksel menggunakan `cv2.VideoCapture()` untuk menangkap gambar secara real-time.

2. Membaca Database Warna

Data dari file DatabaseWarna.txt dibaca menggunakan pandas. File ini berisi nilai RGB dan label warna (Target) yang digunakan untuk pelatihan model.

3. Normalisasi dan Pelatihan Model SVM

Nilai RGB (B, G, R) dinormalisasi menggunakan StandardScaler agar semua fitur memiliki skala yang sama. Model SVM dengan kernel linear dilatih menggunakan data normalisasi dan label warna.

4. **Fungsi Prediksi Warna**

Fungsi predict_color menerima nilai RGB, menormalisasi data input, dan menggunakan model SVM untuk memprediksi warna yang sesuai dengan data yang sudah dilatih.

5. **Loop Kamera untuk Prediksi**

Kamera menangkap frame secara real-time, dan area kecil pada gambar (220:260, 300:340) dianalisis untuk mendapatkan rata-rata nilai RGB. Nilai RGB tersebut dikirim ke fungsi predict_color untuk mendapatkan label warna yang diprediksi. Prediksi warna ditampilkan di jendela kamera dan dicetak ke konsol.

6. **Keluar Program**

Program berjalan hingga tombol ESC ditekan. Setelah itu, kamera dilepaskan, dan semua jendela ditutup.