

**Nama: Agung Suryanto**

**Nim: 230741108**

**Matkul: Kecerdasan Tiruan (UAS)**

## **Penjelasan Object Detection Project**

```
from ultralytics import YOLO # Untuk memuat model YOLO dan melakukan deteksi
import cv2 # Untuk pemrosesan gambar dan video
import streamlit as st # Untuk membuat antarmuka web
from PIL import Image # Untuk manipulasi gambar
import numpy as np # Untuk operasi berbasis array
from collections import Counter # Untuk menghitung jumlah objek yang terdeteksi
```

### **Fungsi:**

- Mengimpor pustaka yang diperlukan untuk memuat model YOLO, membaca data dari kamera, dan menampilkan hasil deteksi di aplikasi Streamlit.

---

### **2. Fungsi load\_model**

python

Salin kode

```
@st.cache_resource
```

```
def load_model(model_path):
```

```
    return YOLO(model_path)
```

### **Penjelasan:**

- **@st.cache\_resource:** Cache hasil dari fungsi load\_model agar model hanya dimuat satu kali. Ini meningkatkan efisiensi.
- **YOLO(model\_path):** Memuat model YOLO dari file yang ditentukan oleh model\_path.

---

### **3. Fungsi display\_results**

python

Salin kode

```

def display_results(image, results):

    boxes = results.boxes.xyxy.cpu().numpy() # Koordinat bounding box
    scores = results.boxes.conf.cpu().numpy() # Skor kepercayaan
    labels = results.boxes.cls.cpu().numpy() # Indeks kelas
    names = results.names # Nama kelas objek

    detected_objects = []

    for i in range(len(boxes)):
        if scores[i] > 0.5: # Ambang batas kepercayaan
            x1, y1, x2, y2 = boxes[i].astype(int)
            label = names[int(labels[i])]
            score = scores[i]
            detected_objects.append(label)
            cv2.rectangle(image, (x1, y1), (x2, y2), (0, 255, 0), 2) # Gambar kotak
            cv2.putText(image, f'{label}: {score:.2f}', (x1, y1 - 10),
cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 255, 0), 2) # Teks label

    return image, detected_objects

```

#### Penjelasan:

- Mengambil hasil deteksi (bounding box, skor, label) dari model YOLO.
- Gambar kotak di sekitar objek yang terdeteksi pada gambar.
- Mengembalikan gambar yang sudah diberi kotak dan daftar objek yang terdeteksi.

## 4. Fungsi main

python

Salin kode

```

def main():

    st.title("Real-time Object Detection with YOLO")

    st.sidebar.title("Settings")

```

```

model_path = "yolo11n.pt" # Path ke model YOLO
model = load_model(model_path)

# Tombol checkbox untuk memulai/menghentikan deteksi objek
run_detection = st.sidebar.checkbox("Start/Stop Object Detection",
key="detection_control")

# Jika checkbox diaktifkan
if run_detection:
    cap = cv2.VideoCapture(0) # Buka kamera
    st_frame = st.empty() # Placeholder untuk frame video
    st_detection_info = st.empty() # Placeholder untuk informasi deteksi

    while True:
        ret, frame = cap.read()
        if not ret:
            st.warning("Failed to capture image.")
            break

        frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB) # Konversi ke RGB untuk
Streamlit
        results = model.predict(frame, imgsz=640) # Deteksi objek

        frame, detected_objects = display_results(frame, results[0]) # Proses hasil deteksi

        st_frame.image(frame, channels="RGB", use_column_width=True) # Tampilkan
frame
        if detected_objects:
            object_counts = Counter(detected_objects)
            detection_info = "\n".join([f'{obj}: {count}' for obj, count in
object_counts.items()])
        else:

```

```
detection_info = "No objects detected."
```

```
st_detection_info.text(detection_info) # Tampilkan informasi deteksi
```

```
if not st.session_state.detection_control: # Hentikan jika checkbox dimatikan  
    break
```

```
cap.release()
```

### Penjelasan:

- **Tujuan utama:** Membuat antarmuka web untuk mendeteksi objek secara real-time menggunakan kamera.
- Jika tombol checkbox diaktifkan, aplikasi:
  - Membuka kamera.
  - Melakukan deteksi dengan YOLO untuk setiap frame video.
  - Menampilkan hasil deteksi (kotak, label, informasi) di antarmuka web.
- Jika checkbox dimatikan, aplikasi berhenti.

---

## 5. Pemanggilan main

python

Salin kode

```
if __name__ == "__main__":  
    main()
```

### Penjelasan:

- Mengeksekusi fungsi main saat skrip dijalankan sebagai file utama.
- Pastikan typo `_name_` diperbaiki menjadi `__name__`, dan `_main_` menjadi `__main__`.

---

## Kesimpulan

Kode ini adalah aplikasi Streamlit untuk deteksi objek real-time menggunakan YOLO. Pengguna dapat mengaktifkan/mematikan deteksi melalui antarmuka web, melihat hasil deteksi dalam video, dan mendapatkan informasi tentang objek yang terdeteksi.



