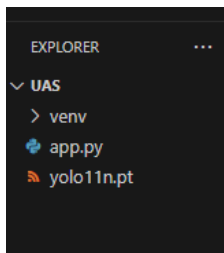


Nama : Efriza Anggi
NIM : 230741116
Prodi/Fakultas : Ilmu Komputer, Teknik dan Sains
Mata Kuliah : Kecerdasan Buatan / Artificial Intelligent (AI)
Dosen Pengampu : Zikri Wahyuzi

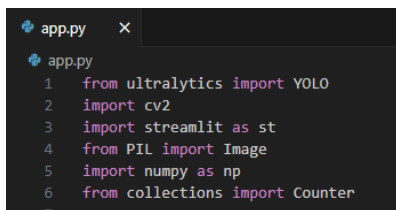
TUTORIAL DAN PENJELASAN MENJALANKAN APLIKASI YOLO DETEKSI

Langkah 1 Mempersiapkan File



- Sebelum memulai, buatlah folder dengan nama yang bebas.
- Pastikan anda telah memiliki program untuk menjalankan python dan pip.
- Siapkan virtual environment dengan “python -m venv venv”
- Silahkan unduh 2 file yang telah disediakan seperti di gambar atas, lalu simpanlah di dalam folder yang telah dibuat.

Langkah 2 Mengunduh Kebutuhan



- Jalankan virtual environment pada terminal dengan “.\venv\Scripts\activate”
- Unduh kebutuhan pustaka sesuai dengan skrip, daftar kebutuhan ada pada gambar di atas. Untuk mengunduh silahkan buka terminal lalu ketik
“pip install ultralytics opencv-python-headless streamlit pillow numpy”

```
1) from ultralytics import YOLO
```

Dari perpustakaan ultralytics, ambil modul YOLO.

a) Ultralytics

Pustaka Python resmi yang dikembangkan oleh YOLO. Pustaka ini menyediakan implementasi cepat dan mudah untuk mendeteksi objek secara Real-Time dengan berbagai model.

b) You Only Look Once (YOLO)

Salah satu model dari ultralytics untuk mendeteksi objek dan segmentasi gambar yang populer, dikembangkan oleh Joseph Redmon dan Ali Farhadi di Universitas Washington. YOLO Memiliki banyak versi, salah satu yang terbaru saat ini adalah YOLO 11 yang menawarkan kinerja canggih di berbagai tugas, termasuk deteksi, segmentasi, estimasi pose, pelacakan, dan klasifikasi, yang meningkatkan kemampuan di berbagai aplikasi dan domain AI.

2) `import cv2`

Ambil modul cv2.

a) Cv2

Modul python dari Open Source Computer Vision Library (OpenCV) yang digunakan untuk berbagai aplikasi pemrosesan citra dan visi komputer. OpenCV adalah pustaka yang sangat populer dalam pengolahan gambar, video, dan deteksi objek secara real-time. Beberapa fungsi utama dari cv2 yaitu Pemrosesan Gambar, Pengolahan Video, Deteksi Objek dan Pengenalan, dan Pengenalan Teks dan Fitur.

3) `import streamlit as st`

Ambil modul streamlit, lalu deklarasikan sebagai st.

a) Streamlit

Pustaka Python yang memungkinkan Anda untuk membuat aplikasi web interaktif dengan cepat dan mudah, terutama untuk aplikasi analisis data, pembelajaran mesin, dan visualisasi data. Beberapa fungsi utama dari streamlit yaitu Membuat Antarmuka Pengguna, Visualisasi Data, dan Komunikasi Real-Time.

4) `from PIL import Image`

Dari pustaka PIL, ambil modul `Image`.

a) Python Imaging Library (PIL)

PIL atau dikenal sekarang sebagai Pillow adalah pustaka untuk memanipulasi gambar. Fungsi Utamanya yaitu membuka, memproses, dan menyimpan file gambar dalam berbagai format (JPEG, PNG, GIF, dll.), serta

Commented [RX1]: l

mendukung operasi manipulasi gambar seperti rotasi, cropping, scaling, dan filter.

```
5) import numpy as np
```

Ambil modul numpy, lalu deklarasikan sebagai np.

a) NumPy

Pustaka Python yang menyediakan dukungan untuk array multidimensi (seperti matriks) dan berbagai fungsi matematis dan statistik yang efisien untuk operasi numerik. Beberapa fungsi utama dari NumPy yaitu Array, Fungsi Matematika, dan Aljabar Linier.

```
6) from collections import Counter
```

Dari Pustaka collections, ambil Subclass Counter.

a) Counter

Subclass dari dictionary yang dirancang khusus untuk menghitung jumlah kemunculan elemen dalam sebuah iterable (seperti list, string, atau tuple).

Langkah 3 Jalankan Skrip

- a. Tetap pada mode virtual environment seperti Langkah 2 bagian a.
- b. Ketik “streamlit run app.py”

Kode tersebut meminta streamlit untuk menjalankan file app.py

Adapun beberapa penjelasan dari skrip app.py yaitu:

```
# Load YOLO model
@st.cache_resource
def load_model(model_path):
    return YOLO(model_path)
```

- 1) @st.cache_resource, untuk menyimpan hasil fungsi di cache agar objek tidak perlu dimuat ulang jika sudah pernah ada.
- 2) load_model(model_path), untuk menjalankan model YOLO dari model_path.

```
# Process and display the detection results
def display_results(image, results, confidence_threshold=0.5):
    boxes = results.boxes.xyxy.cpu().numpy() # [x1, y1, x2, y2]
    scores = results.boxes.conf.cpu().numpy() # Confidence scores
    labels = results.boxes.cls.cpu().numpy() # Class indices
    names = results.names # Class names

    detected_objects = []

    for i in range(len(boxes)):
        if scores[i] > confidence_threshold: # Confidence threshold
            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)
            cv2.putText(image, f"{label}: {score:.2f}", (x1, y1 - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 255, 0))

    return image, detected_objects
```

1) Fungsi display_results(image, results, confidence_threshold=0.5)

Fungsi ini memproses hasil deteksi dari model YOLO berupa parameter results yang akan dilakukan pengulangan sebanyak box yang diberikan untuk diberikan label, dan skor kepercayaan pada tiap objek yang ditemukan dengan persyaratan confidence_threshold > 0.5. Objek yang lolos akan diberikan label dan kotak yang akan disimpan di variabel detected_objects. Fungsi ini mengembalikan parameter image dan hasil objek yang lolos melalui proses pengulangan.

```
# Main Streamlit app
def main():
    st.set_page_config(page_title="YOLO 11: Deteksi Objek", layout="wide", page_icon="🚗")
    st.title("🚗 *YOLO 11*: Real-time Object Detection 🚗 Oleh: Efriza")
    st.sidebar.title("⚙️ *Pengaturan*")

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

    # Provide options: (Detection Camera, Upload Image)
    mode = st.sidebar.radio("Pilih Mode Mendeteksi", ("Real-Time Camera", "Unggah Gambar"))

    # Sidebar: Atur confidence threshold
    confidence_threshold = st.sidebar.slider("Confidence Threshold:", 0.1, 1.0, 0.5, 0.05)

    if mode == "Real-Time Camera":
        st.subheader("🎥 *Mulai Mendeteksi*")
        run_detection = st.sidebar.checkbox("Mulai Deteksi", key="detection_control")
        # Open video capture if checkbox is active
        if run_detection:
            cap = cv2.VideoCapture(0) # Open Camera
            st_frame = st.empty() # Placeholder for video frames
            st_detection_info = st.empty() # Placeholder for detection information

            while True:
                ret, frame = cap.read()
                if not ret:
                    st.warning("❌ Gagal dalam menangkap gambar dari kamera.")
                    break

            # Run YOLO detection
```

1) Fungsi main()

Pada fungsi ini dilakukannya konfigurasi terhadap streamlit yang akan ditampilkan, misalnya "set_page_config", "title", dan "sidebar.title". Selain itu, skrip akan membaca model yang telah ditentukan yaitu "yolo11n.pt". Selanjutnya, skrip akan menambahkan opsi/pilihan pada sidebar berupa "radio" dengan pilihan

“Real-Time Camera” dan “Unggah Gambar” serta menambahkan `confidence_threshold`.

Ketika pengguna memilih mode “Real-Time Camera”, skrip akan memproses pengambilan data pada kamera yang akan filtering dengan persyaratan pengambilan data berhasil dibaca. Selanjutnya skrip akan memprediksi dengan model yang ada dan diteruskan ke fungsi `display_results` dari hasil prediksi. Hasil dari fungsi `display_results` akan diproses untuk ditampilkan pada pengguna.

Ketika pengguna memilih mode “Unggah Gambar”, skrip akan menampilkan lama pengunggahan yang jika pengguna mengunggah akan diproses prediksi lalu dibaca oleh fungsi `display_results` diakhiri dengan menampilkan hasil.

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

- 1) Skrip di atas untuk menjalankan fungsi `main()`

Langkah 4 Akses Localhost

```
PS C:\Users\RINBL\uas> .\env\Scripts\activate  
(venv) PS C:\Users\RINBL\uas> streamlit run app.py  
  
You can now view your Streamlit app in your browser.  
  
Local URL: http://localhost:8501  
Network URL: http://192.168.1.6:8501  
  
2024-12-24 23:18:42.566 Examining the path of torch.classes raised: Tried to instantiate class '___path___path', but it does not exist! Ensure tha  
t it is registered via torch::class_
```

- a. Setelah streamlit aktif seperti di atas, silahkan akses link yang tertera
- b. Terdapat 2 pilihan sesuai dengan skrip, yaitu Real-Time Camera dan Upload Image

- 1) Real-Time Camera

Silahkan check pada Mulai Deteksi lalu atur `confidence_threshold` sesuai keinginan. Streamlit akan memproses masukan lalu mengeluarkan hasil pada sebelahny.

- 2) Unggah Gambar

Silahkan unggah gambar yang akan dideteksi dan hasil akan keluar di sebelahny.