

Nama : Alfian Asa Ramadhoni

Prodi : Ilmu Komputer

Matkul : UAS Kecerdasan Tiruan

PENJELASAN KODE SERTA TUTORIAL UNTUK MENDEPLOY STREAMLIT

a). Penjelasan Kode

1). Pertama tama kita harus mengimport atau menginstall library yang akan kita gunakan untuk menjalankan streamlit maupun fungsi yang lainnya seperti ada pada gambar ini saya menggunakan ultralytics, cv2, streamlit, PIL, numpy, collections, dan base 64

```
from ultralytics import YOLO
import cv2
import streamlit as st
from PIL import Image
import numpy as np
from collections import Counter
import base64
```

2). Nah disini saya langsung menambahkan fungsi untuk latar belakang/background. Ini digunakan untuk menambah gambar maupun video pada latar belakang streamlit

```
# Fungsi untuk menambahkan latar belakang
def set_background(image_path):
    with open(image_path, "rb") as file:
        base64_image = base64.b64encode(file.read()).decode()
    css = f"""
    <style>
    .stApp {{
        background-image: url("data:image;base64,{base64_image}");
        background-size: cover;
        background-repeat: no-repeat;
        background-attachment: fixed;
    }}
    </style>
    """
    st.markdown(css, unsafe_allow_html=True)
```

3). Setelah itu kita load YOLO model agar menyimpan hasil pemanggilan fungsi sehingga tidak perlu dimuat ulang pada setiap interaksi pengguna.

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

4). Nah ini adalah fungsi yang bertugas untuk memproses dan menampilkan hasil deteksi objek pada gambar atau frame yang diberikan, yang dihasilkan oleh model YOLO.

```
# Function untuk memproses hasil deteksi
def display_results(image, results):
    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] > 0.5: # 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), 2)

    return image, detected_objects
```

5). Fungsi ini adalah fungsi utama atau main function dimana seluruh fungsi terletak pada def main(): seperti Markdown yang saya gunakan untuk judul pada halaman utama streamlit, ini menggunakan perpaduan CSS serta sidebar

```
# Fungsi utama aplikasi
def main():

    # Menambahkan teks "Real-time Object Detection with YOLO"
    st.markdown("<h1 style='text-align: center;'>Real-time Object Detection with YOLO - ALFAN</h1>", unsafe_allow_html=True)

    # Menambahkan judul pada sidebar
    st.sidebar.title("SELAMAT DATANG DI REAL TIME OBJECT DETECTION")
```

6). Fungsi ini digunakan untuk menambah subheader, teks input, select box, serta button pada sidebar streamlit seperti yang sudah saya jelaskan pada gambar ini

```
# Menambahkan subheader pada side bar
st.sidebar.subheader("Lengkapi Data Diri")

# Menambahkan teks input atau mengetikkan teks
st.sidebar.text_input('Nama Lengkap')

# Menambahkan selectbox sebagai pilihan
option = st.sidebar.selectbox(
    'Jenis Kelamin',
    ('Laki-Laki', 'Perempuan'),
)

# Menambahkan fungsi umur dalam bilangan bulat
umur = st.sidebar.number_input('Umur', min_value=0, max_value=120, step=1, format="%d")

# Menambahkan button atau Tombol
st.sidebar.button('simpan')
```

7). kode ini bertujuan untuk mengatur alur deteksi objek dengan webcam atau camera menggunakan button untuk memulai dan menghentikan deteksi, serta memuat model YOLO yang digunakan untuk mendeteksi objek dalam frame camera.

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

# Tambahkan dua tombol untuk Start dan Stop
start_button = st.sidebar.button("Start Object Detection")
stop_button = st.sidebar.button("Stop Object Detection")

# Menggunakan flag untuk mengontrol deteksi
detection_active = False

if start_button:
    detection_active = True

if stop_button:
    detection_active = False

if detection_active:
    cap = cv2.VideoCapture(0)
    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("Failed to capture image.")
            break
```

8). kode ini bertanggung jawab untuk menampilkan hasil deteksi objek pada feed video real-time, memperbarui informasi deteksi yang ditampilkan, dan memungkinkan deteksi objek berjalan secara dinamis berdasarkan kontrol dari pengguna.

```
# Run YOLO detection
frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB) # Convert to RGB for display
results = model.predict(frame, imgsz=640) # Perform detection

# Draw results and collect detected objects
frame, detected_objects = display_results(frame, results[0])

# Display video feed
st_frame.image(frame, channels="RGB", use_column_width=True)


# Display detection information
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) # Update detection info text

# Break the loop if detection is turned off
if not detection_active:
    break
```

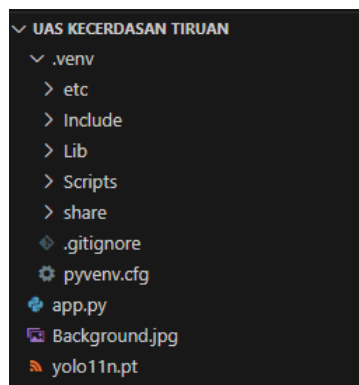
B). Tutorial Mendeploy

1). Pastikan sudah menginstall virtual env (venv), jika belum silahkan di install terlebih dahulu

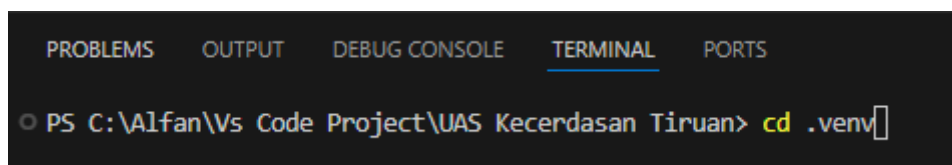
Untuk menginstal virtual environment (venv) Python, Anda dapat melakukan langkah-langkah berikut: 

1. Buka Terminal
2. Ketikkan perintah `python3 -m pip install --user virtualenv`

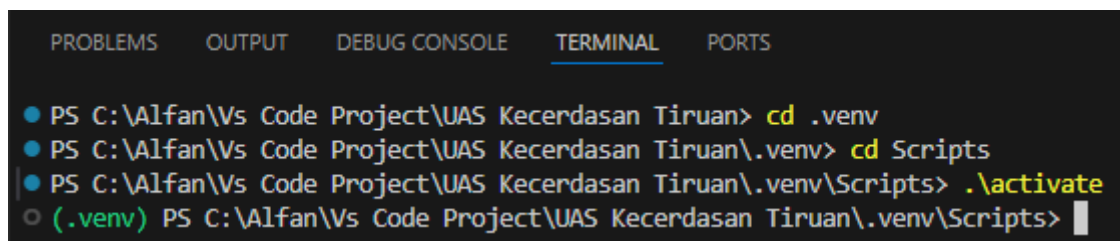
2). Jika venv nya sudah terinstall, di dalam project akan muncul direktori Bernama .venv seperti pada Digambar ini



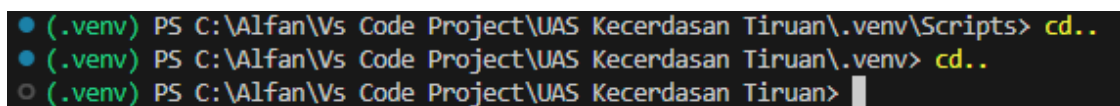
3). Setelah itu bisa ketikan `cd .venv` seperti di gambar berikut



4). Setelah itu bisa lanjut dengan memasukan kode `cd Scripts` kemudia `.\activate`. Setelah memasukkan kode tersebut, kita sudah bisa mengakses venv (virtual env) seeperti pada Digambar



5). Setelah masuk ke .venv kita Kembali ke direktori awal dengan memasukkan kode `cd..`.



6). Setelah itu kita lanjut memasukkan kode streamlit run app.py dan akan mengenerate suatu web local atau local host

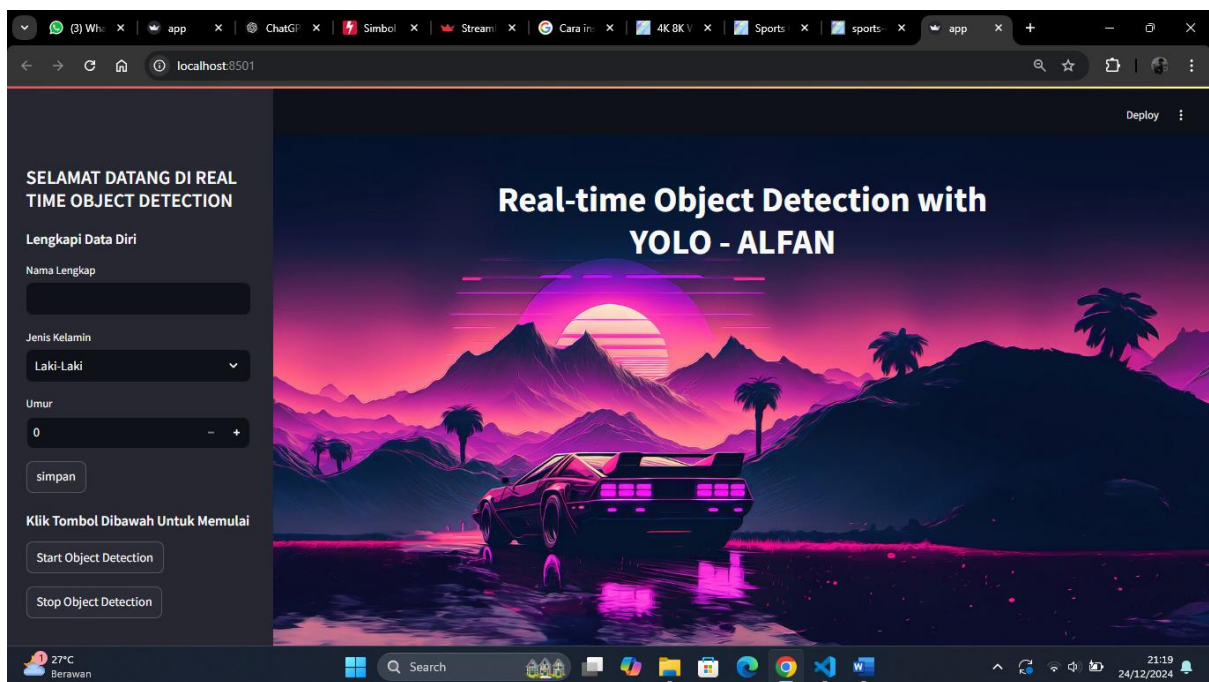
```
(.venv) PS C:\Alfan\Vs Code Project\UAS Kecerdasan Tiruan> streamlit run app.py

You can now view your Streamlit app in your browser.

Local URL: http://localhost:8501
Network URL: http://192.168.1.16:8501

2024-12-24 21:16:51.362 Examining the path of torch.classes raised: Tried to instantiate class '__path__._path', but it does not exist! Ensure th
at it is registered via torch::class_
```

7). Tampilan web dari streamlit sudah berhasil kita buka dan akan tampil deploy dari hasil kode yang telah kita buat seperti ini

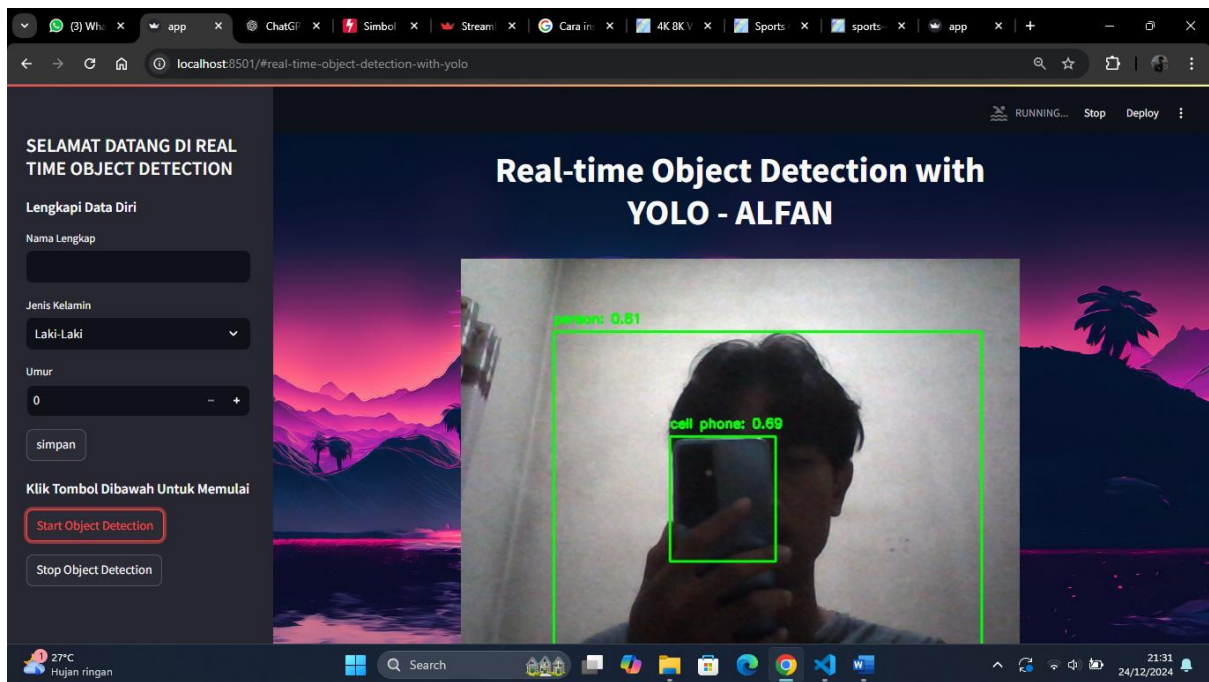


*Ini adalah tampilan web local host streamlit yang sudah berhasil saya buat

8). Pada halaman atau pada sidebar ini saya membuat sebuah interaksi seperti form yang dapat diisi seperti Nama Lengkap, Jenis Kelamin, dan Umur

A close-up screenshot of the 'Lengkapi Data Diri' form in the Streamlit application. The form is on a dark background. It contains three input fields: 'Nama Lengkap' with the text 'Alfan Asa Ramadhoni', 'Jenis Kelamin' with a dropdown menu showing 'Laki-Laki', and 'Umur' with a numeric input field showing '19'. Below these fields is a 'simpan' button.

9). Nah ini adalah button dimana kita akan menjalankan Real Time Object Detection, tekan pada button “Start Object Detection” dan selamat kita sudah berhasil untuk mendeteksi sebuah objek



10). Untuk menonaktifkannya cukup tekan saja button “Stop Object Detection” dan jangan lupa untuk save atau simpan projek teman teman

Link Github : <https://github.com/alfanasarmdhn/UAS-Kecerdasan-Tiruan>