

Name: Yash Gaikwad
Roll No: 381071
PRN: 22420141

Assignment 7

Problem Statement: Object detection using YOLO and Pretrained Model.

Introduction:

Object detection is a fundamental task in computer vision that involves identifying and localizing multiple objects within an image or video. Unlike image classification, which only predicts a single label for an image, object detection not only classifies objects but also provides bounding boxes around them.

In this assignment, we use **YOLO (You Only Look Once)**, one of the fastest and most accurate object detection algorithms, along with a **pretrained model**. Pretrained models leverage large datasets such as COCO or Pascal VOC, enabling faster development and higher accuracy without requiring training from scratch.

Objective:

The objectives of this assignment are:

- To understand the concept of object detection and its importance in computer vision.
- To implement object detection using the **YOLO algorithm**.
- To utilize a **pretrained YOLO model** for detecting multiple objects in real-time images or videos.
- To evaluate the accuracy and speed of YOLO compared to traditional detection techniques.

Theory:

1 Object Detection

- Combines **classification** (what the object is) and **localization** (where the object is).
- Applications include autonomous driving, surveillance, robotics, medical imaging, and smart retail.

2 YOLO (You Only Look Once)

- YOLO is a **real-time object detection algorithm**.
- Unlike traditional methods (like R-CNN) which perform region proposals and classification separately, YOLO frames object detection as a **single regression problem**.
- The image is divided into an **$S \times S$ grid**, and each grid cell predicts:
 1. Bounding box coordinates (x, y, width, height).
 2. Objectness score (confidence of an object being present).
 3. Class probabilities.

Advantages of YOLO:

- Extremely fast (real-time detection).
- Unified architecture (single neural network).
- Generalizes well to unseen data.

3 Pretrained Models

- Training YOLO from scratch requires huge datasets and computational resources.
- Instead, pretrained weights trained on large datasets like **COCO (Common Objects in Context)** can be used.
- Pretrained YOLO versions include **YOLOv3, YOLOv4, YOLOv5, YOLOv7, and YOLOv8**, each improving accuracy and speed.

4 Workflow for Object Detection using YOLO

1. **Image/Video Input** – Load the input source.
2. **Preprocessing** – Resize and normalize the input image.
3. **YOLO Model Inference** –
 - Pass the input through the pretrained YOLO network.
 - Predict bounding boxes, confidence scores, and class probabilities.
4. **Post-processing** – Apply **Non-Maximum Suppression (NMS)** to remove overlapping boxes.
5. **Output** – Draw bounding boxes and labels on detected objects in the image/video.

Conclusion:

In this assignment, we implemented **object detection using YOLO with a pretrained model**. YOLO's unified architecture allowed us to perform detection in real-time, making it highly suitable for practical applications. By leveraging pretrained weights, we achieved accurate object detection without requiring extensive training. This project highlights the effectiveness of YOLO in handling object detection tasks and provides a strong foundation for extending to advanced applications such as video analytics, autonomous systems, and real-time surveillance.