Q2. YOLOv8 Real-Time Object Detection Report

Implementation Approach:

- Utilized Ultralytics YOLOv8 (yolov8s.pt) for real-time object detection.
- Captured live video using OpenCV.
- Loaded YOLOv8 model via YOLO("yolov8s.pt").
- Per-frame inference, drew bounding boxes with class labels and confidence scores.
- Displayed results with cv2.imshow().

Results & Performance Analysis:

- Achieved real-time inference (20-30 FPS) on webcam input using YOLOv8s.
- Detected common COCO classes with high accuracy and low latency.
- Smooth execution on x86_64 and ARM platforms with GPU or CPU

Challenges Faced & Solutions:

1. Torch 2.6+ security change broke model loading -
Solution: Downgraded to torch==2.5.1 for backward compatibility.

2. Version conflicts with torchvision -
Fixed using versions: -
torch==2.5.1
torchvision==0.16.1
torchaudio==2.5.1

3. Matplotlib warning about Axes3D -
Caused by multiple installs or missing optional 3D backend - **not critical.**

Future Improvements -

1. Add GPU acceleration check for performance boost on supported devices.
2. Switch to yolov8n.pt for faster speed on low-power systems.
3. Add options to record video, count detections, or trigger alerts.
4. Expand to custom-trained YOLOv8 models for specialized domains.

5. Package code in Docker or provide cross-platform setup scripts.