Rishi Shanthan Bhagavatham(50560689) – Meenakshi Gundluri(50568942)

Title: YOLO-based Real-Time Object Detection in Video Streams

Project Overview

This project focuses on implementing a real-time object detection system in video streams using the YOLO (You Only Look Once) algorithm. The application will demonstrate YOLO's capabilities to process and identify objects efficiently in video frames, making it suitable for tasks such as surveillance, autonomous navigation, and video content analysis.

Application Purpose: The goal is to develop a tool that can accurately detect and classify objects in real-time, benefiting industries like transportation, security, and content creation. This solution is particularly advantageous for scenarios requiring high-speed and accurate object recognition.

State of the Art

Object detection has seen significant advancements with YOLO models being state-of-the-art for real-time applications. Existing solutions include:

1. **YOLOv3, YOLOv4, and YOLOv5**: These models offer a balance between speed and accuracy.
2. **Other Approaches**: Methods like Faster R-CNN and SSD (Single Shot Detector) provide high accuracy but fall short in real-time processing capabilities compared to YOLO.

This project leverages the strengths of YOLOv3 for its optimal performance.

Inputs and Outputs

- **Inputs**: Video streams or pre-recorded videos.
- **Outputs**: Annotated video streams where detected objects are labeled with bounding boxes and class names.

System Diagram:

1. Input video frame.
2. YOLO model processes each frame.
3. Output video frame with bounding boxes and labels.

Data

- **Darknet:** In this project we will be using darknet architecture that is pre-trained on Microsoft Coco dataset which can identify upto 80 catagories. (https://github.com/pjreddie/darknet)

- **Dataset**: COCO (Common Objects in Context) dataset, widely used for object detection tasks.
- **Acquisition**: The dataset is publicly available and will be utilized for training and evaluation.
- **Scale**: Thousands of labeled images across 80 classes.

Coding Resource Requirements

- Libraries: OpenCV, YOLO model weights (provided in darknet architecture).
- Online Resources: Pre-trained YOLO weights from public repositories. (provided in darknet architecture)
- Independent Contributions: Custom preprocessing and postprocessing pipelines for video handling and evaluation metrics.

Computational Resources

- **Hardware**: A GPU-enabled system for training and inference.
- **Effort**: Estimated 100 person-hours.
- **Duration**: Training and testing phases span approximately two weeks.

Evaluation

- **Success Criteria**: High accuracy (mAP - mean Average Precision) on object detection tasks.
- **Metrics IoU**:
    - dhaka_traffic.mp4 – 0.3
    - road_traffic.mp4 – 0.6
    - road_traffic.mp4 – 0.8
- **Test Scenarios**: Positive cases with high detection rates and challenging cases with occlusions or low lighting conditions.

Project Expectations

Excitement: Leveraging cutting-edge technology to solve real-world problems. **Learning Goals**: Deepen understanding of computer vision techniques, particularly YOLO's architecture and optimization strategies.

Results and Analysis

The project will showcase:

- Visual outputs of object detection in various scenarios.
- Comparison with existing YOLO implementations and analysis of limitations.

Bibliography

- SK Mansoub, R Abri, A Yarıcı - SIGNAL, 2019 - personales.upv.es

- You Only Look Once: Unified, Real-Time Object Detection - Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi

- Redmon, J., et al., "YOLO9000: Better, Faster, Stronger."
- Bochkovskiy, A., et al., "YOLOv4: Optimal Speed and Accuracy of Object Detection."
- Real-Time Object Detection for Streaming Perception - *Jinrong Yang, Songtao Liu, Zeming Li, Xiaoping Li, Jian Sun*; Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 5385-5395