SafeRoads: AI-Enabled Accident Detection (Team Quantum Quotients)

Goal: Develop an AI-driven system called SafeRoads to detect accidents in CCTV footage in real-time.

Github Link:
https://github.com/Puyush/Accident_Detection_Yolov8

Live:
https://universe.roboflow.com/puyush-fipgg/real-time-accident-detection/model/1

This report aims to provide a comprehensive overview of our implementation of the YOLO-v8 model for real-time object detection in COMSYS Hackathon-3 which revolutionized object detection by achieving high accuracy while maintaining remarkable processing speed.

YOLOv8 Architecture:

YOLOv8 architecture is built upon the foundation of deep convolutional neural networks (CNNs), specifically designed for real-time object detection. It integrates features from its predecessors while introducing enhancements to improve detection accuracy and efficiency. There are 3 essential blocks in the YOLO algorithm which are: Backbone, Neck and Head.

Backbone:

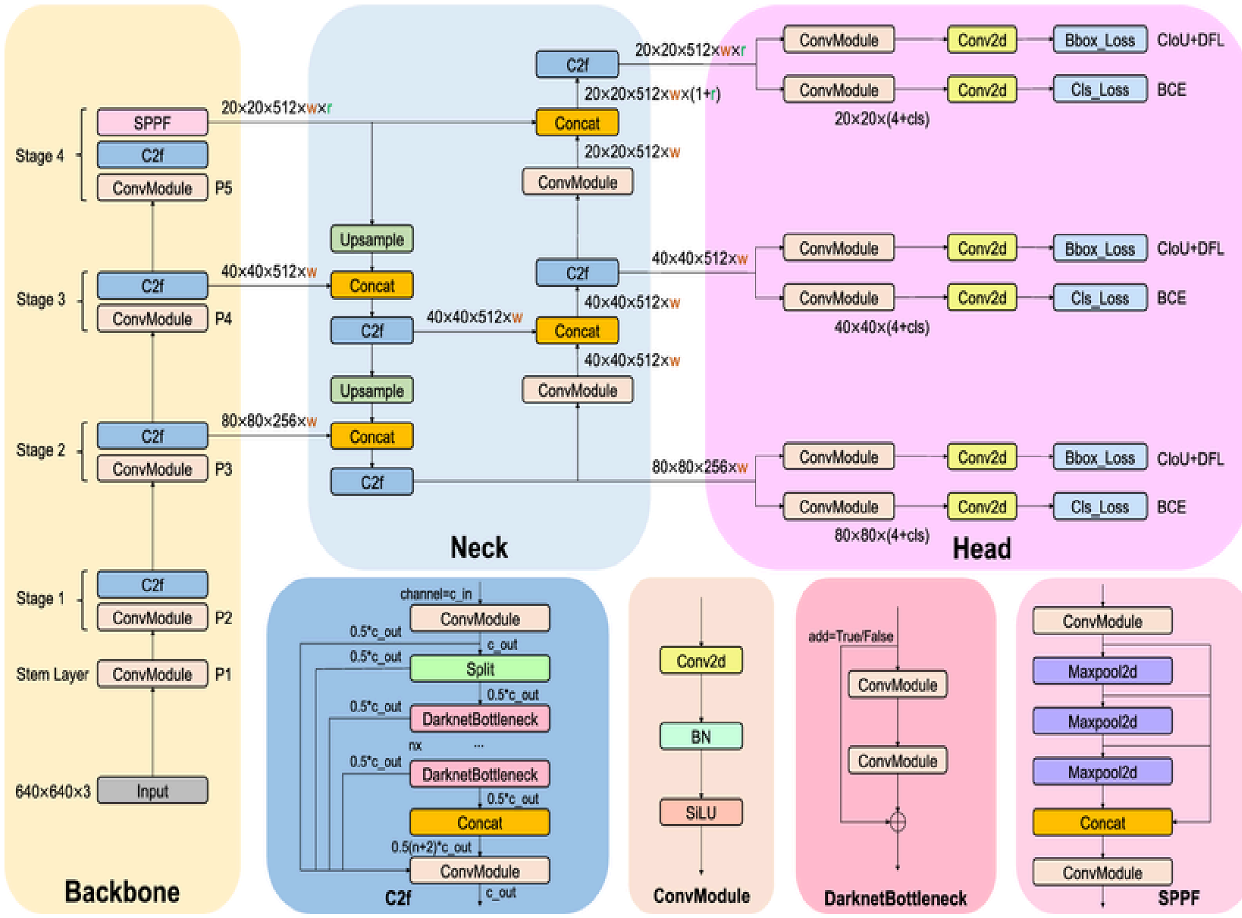
The backbone, also known as the feature extractor, is responsible for extracting meaningful features from the input. It captures simple patterns in the initial layers, such as edges and textures. A modified version of the CSPDarknet53 architecture forms the backbone of YOLOv8. This architecture consists of 53 convolutional layers and employs cross-stage partial connections to improve information flow between the different layers.

Neck:

It performs feature fusion operations and integrates contextual information. Basically, Neck assembles feature pyramids by aggregating feature maps obtained by the Backbone. The fusion of features from different scales equips YOLOv8 with the versatility to detect objects of varying sizes and aspect ratios, ensuring robust performance across a spectrum of scenarios. It reduces the spatial resolution and dimensionality of resources to facilitate computation, a fact that increases speed but can also reduce the quality of the model.

Head:

The detection head of YOLOv8 consists of multiple convolutional layers responsible for predicting bounding boxes, confidence scores, and class probabilities for objects within the input image. This component utilizes a single unified architecture to simultaneously perform object localization and classification.



Loss Function:

$$
\lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right]
$$

$$
+ \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right]
$$

$$
+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2
$$

$$
+ \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2
$$

$$
+ \sum_{i=0}^{S^2} \mathbb{1}_{i}^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2
$$

Result:

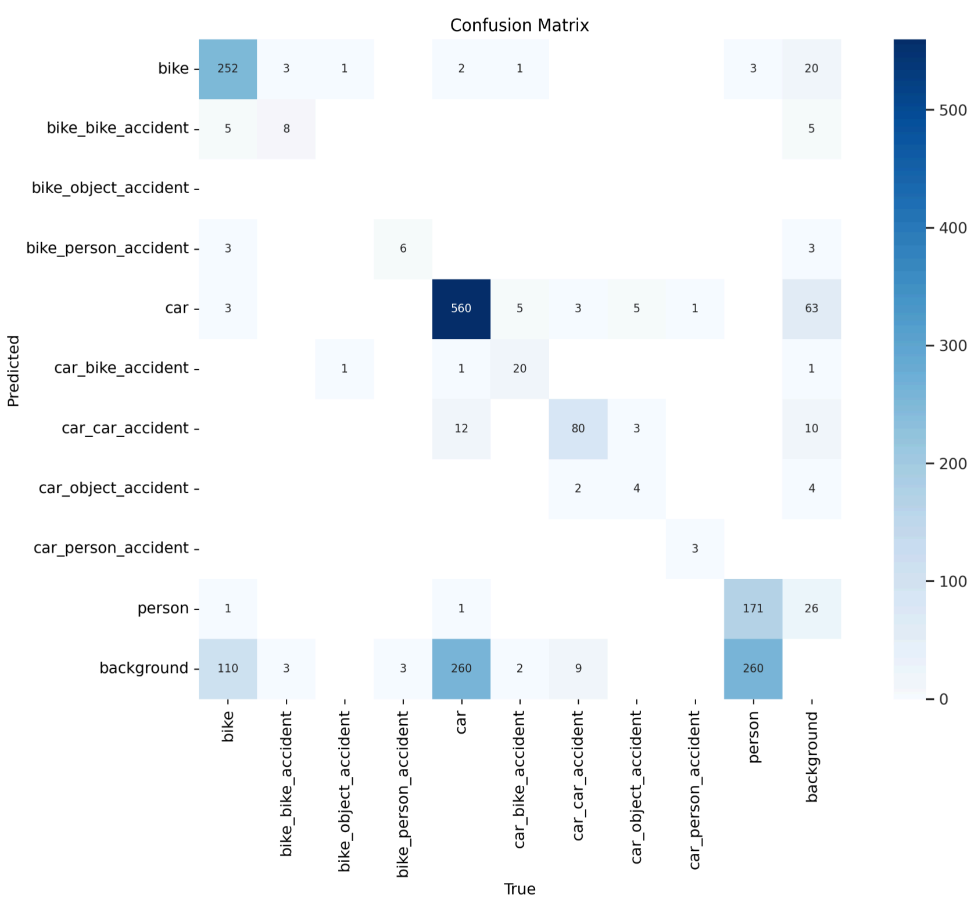

car_bike_accident 0.9
bike_bike_accident 1.0
bike_bike_accident 0.5

On test data:

Total accident images processed: 30
Accident detected in 22 accident images
Accident Detection Accuracy: 73.33 %

Total non-accident images processed: 70
Non-accident detected in 62 non-accident images
Non-Accident Detection Accuracy: 88.57 %

Total images processed: 100
Total correct predictions: 84
Overall Accuracy: 84.0 %



Trade-off between IOU and Confidence Threshold:

The Intersection over Union (IOU) and Confidence Threshold are two critical parameters in object detection algorithms like YOLO, and there exists a trade-off between them:

IOU Threshold (intersection/union): IOU measures the overlap between the predicted bounding box and the ground truth bounding box. Basically, it shows how much two bounding boxes overlap each other. A higher IOU threshold leads to stricter criteria for considering a detection as correct. Setting a high IOU threshold increases the precision of detections but may also result in missed detections, especially in scenarios where objects are partially occluded or overlapping.

Confidence Threshold: The Confidence Threshold determines the minimum confidence score required for considering a detection as valid. Lowering the confidence threshold increases the sensitivity of the model, allowing it to detect more objects, including those with lower confidence scores. However, this also increases the likelihood of false positives and incorrect detections.

Trade-off Explanation:

- **High IOU Threshold, Low Confidence:**

Setting a high IOU threshold and a low confidence threshold favors recall over precision. The model will detect objects with high overlap with ground truth even if they have low confidence scores, potentially capturing more true positives but also increasing the chances of false positives due to the lower confidence threshold.

- **Low IOU Threshold, High Confidence Threshold**:

Conversely, setting a low IOU threshold and a high confidence threshold prioritizes precision over recall. The model will only detect objects with high confidence scores, and even those with lower overlap with ground truth may be disregarded. This approach reduces the number of false positives but may result in missed detections, particularly in challenging scenarios.

In summary, finding the optimal balance between IOU and confidence threshold is crucial in object detection tasks. This balance depends on the specific requirements of the application, the characteristics of the dataset, and the trade-off between precision and recall that is acceptable for the task at hand.

Limitations:

Partial Object Occlusion: YOLO may struggle to accurately detect objects when they are partially occluded by other objects in the scene. If only a small portion of the object is visible due to occlusion, YOLO might fail to recognize it entirely or misclassify it.

Object Stacking and Overlapping: Another limitation of YOLO is its difficulty in accurately detecting individual objects when they are densely packed or stacked on top of each other. For instance, in scenarios like a flock of birds or a dense crowd, YOLO may struggle to distinguish between individual objects, leading to missed detections or incorrect classifications.

Thank You!