COMPUTER VISION

TASK2: OBJECT DETECTION AND DEPTH ESTIMATION

NAME	MATRICULATION NUMBER	STUDY PROGRAMME
Manoj Nagendrakumar	16344060	Master's Mechatronics
Mohammed Kumail Abbas	18743947	Master's Mechatronics

Guided by:

Prof.Dr.Stefan Elser

1 Introduction

The goal of this project is to use 2D object detections from images to estimate the 3D depth of objects, specifically cars, using a subset of the KITTI dataset. Object detection is performed using YOLOv8x trained model and depth estimation is carried out by using intrinsic matrix of the camera setup provided by the KITTI dataset[1]. This report outlines the steps taken, the mathematical approach for depth estimation, the results obtained, and the analysis of discrepancies between estimated distances and ground truth values.

2 Approach

- The algorithm begins by importing the necessary libraries such as os for file system interactions, numpy for numerical operations, cv2 for computer vision tasks, and matplotlib.pyplot for image visualization.

- Detecting cars and drawing bounding boxes with Yolov8x trained model(as this model is most accurate model trained compare to other model), it is easier to draw and save the bounding box coordinates. After deploying the model on a frame, it saves the (x1,y1) and (x2,y2) coordinates to the box list. The two corners of the box can then be easily accessed to draw the bounding box.

- Loading the given ground truth bounding boxes After deploying the Yolo model and obtaining the bounding boxes, the boxes from the ground truth extracted from the labels folder and plotted on the image using computer vision techniques.On comparing the result of given data with output data, it was observed that

the model detects extra cars which are not given in the ground truth.In order to plot the results and calculate the accuracy, the cars in the ground truth and those predicted by the model must be the same.

- Matching Detections with Ground Truth To pair detections with ground truth objects, we compute the Intersection over Union (IoU) between each detected bounding box and the corresponding ground truth boxes. The IoU measures the overlap between two bounding boxes, offering a measure of their similarity. For each detected car, we choose the ground truth box with the highest IoU as the matched object. If the IoU falls below a set threshold(e.g., 0.5) or if no ground truth box is found, the detection is considered as false positive.

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}}$$

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}}$$

In order to achieve this,a algorithm was designed which is IOU (Intersection over union)technique used widely in computer vision which measures the accuracy of the model.The formula of IOU is given below.

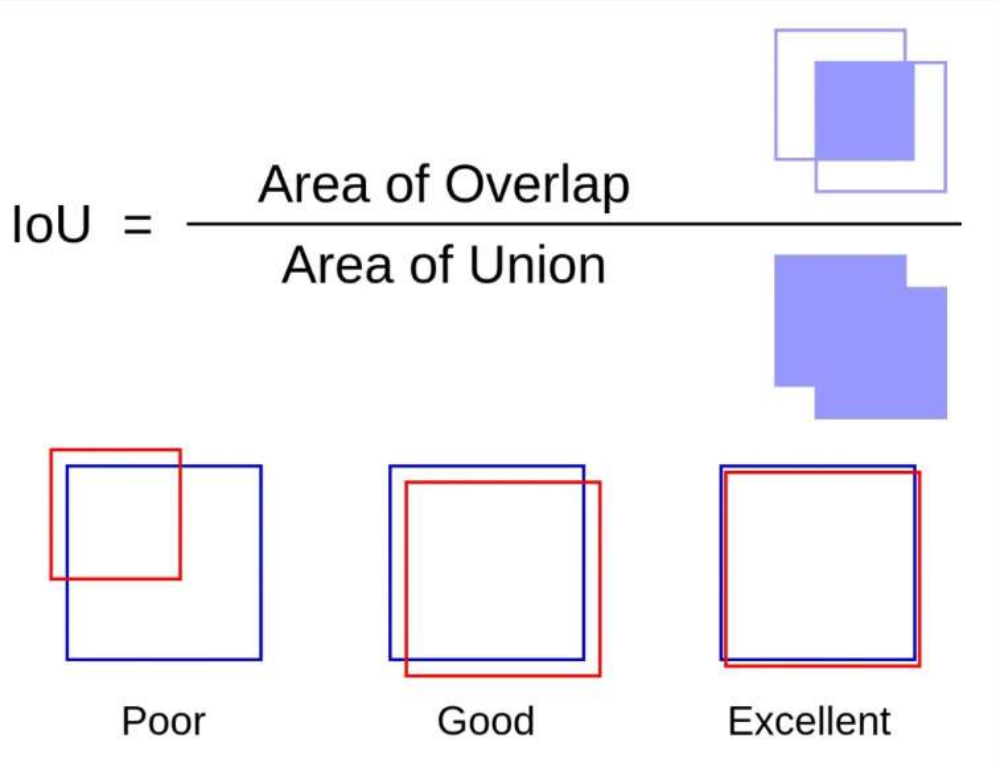

Figure 1: IOU Representation

- Algorithm Implementation for Depth Estimation, the given dataset also contains the intrinsic matrices of each frame in the ground truth. The intrinsic matrix contains the inner parameters of the cameras mounted on vehicle which project all the 3D points in their Field-of-View (FOV) on their 2D image planes[2].

- Calculations

$$\text{Midpoint} = \left(\frac{x_1 + x_2}{2}, y_2\right)$$

Where:(x_1, y_1) = top-left corner (x_2, y_2) = bottom-right corner

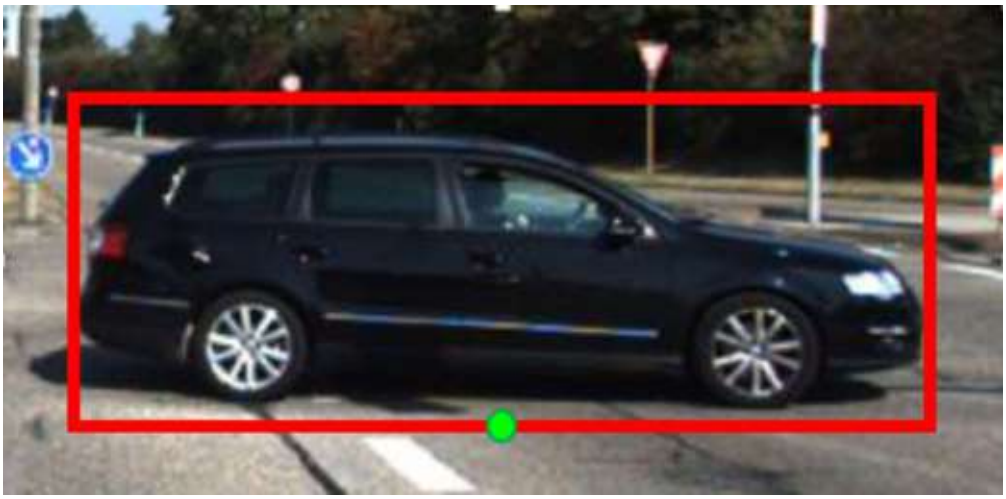

Figure 2: Mid point for depth estimation

$$\mathbf{p} = \begin{bmatrix} x_{\text{new}} \\ y_2 \\ 1 \end{bmatrix}$$

- The *intrinsic matrix* encodes the camera's internal parameters, like focal lengths and optical center:

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

  Where:

  - f_x, f_y: Focal lengths in the x and y directions
  - c_x, c_y: Principal point (optical center)

- To find the direction of the point in the 3D camera coordinate system, the *inverse of the intrinsic matrix* is applied:

$$\mathbf{d} = \mathbf{K}^{-1} \cdot \mathbf{p}$$

$$Z = h/\mathbf{d}_z$$

Where $\mathbf{d}_z$ is the z-component of $\mathbf{d}$.

$$X = Z \cdot \mathbf{d}_x, \quad Y = Z \cdot \mathbf{d}_y$$

- The distance between the camera and the intersection point is calculated using the **Euclidean distance formula**:

$$\text{Distance} = \sqrt{X^2 + Y^2 + Z^2}$$

3 Results

The results showed a range of accuracies in the distance estimations. while many of the calculated distances closely matched the ground truth, there were notable changes in certain cases. Below are the detailed analysis of both successful and complex scenarios.

3.1 Images with proper detection

The below images with clear, unobstructed views of the cars, the distance estimations were highly accurate it is due to following factors such as clear visibility, consistent Lightening and flat Terrain.

Figure 3: Frame Id: 006227

3.2 Complex scenarios

In some images, distance is estimated significantly and some are deviated from the ground truth. several factors contributed to these, which includes Hiding or occlusion of images on one another in this case cars, False positives or incorrect identifying of objects as cars and weather conditions which is affecting object visibility.

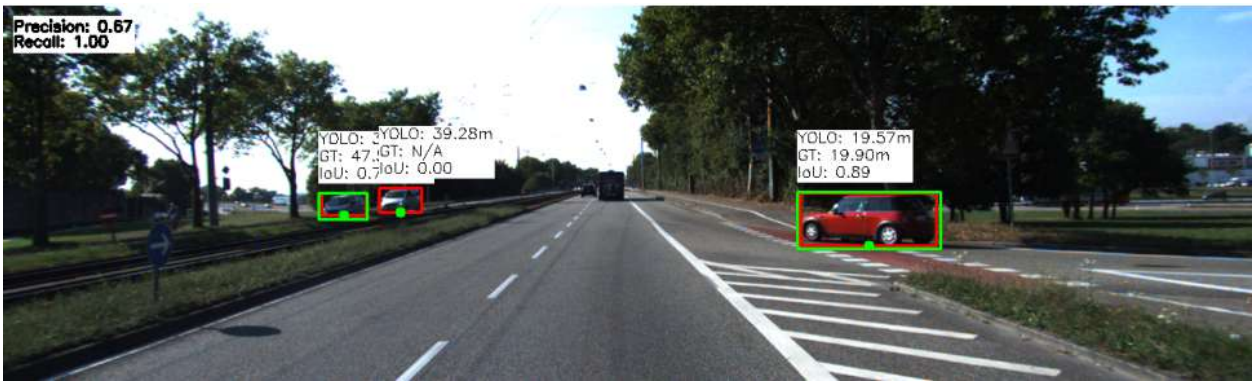

Figure 4: Frame Id: 006042

Figure 5: Frame Id: 006059

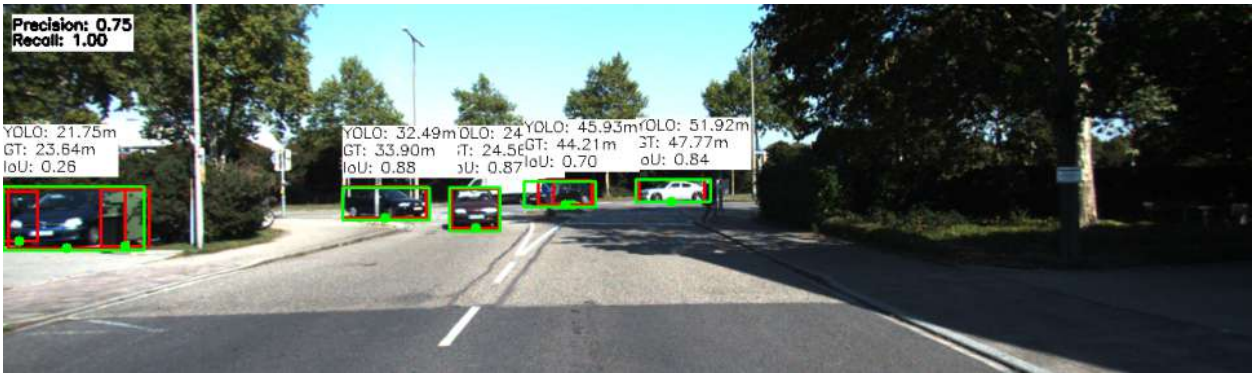

Figure 6: Frame Id: 006067

Figure 7: Frame Id: 006206

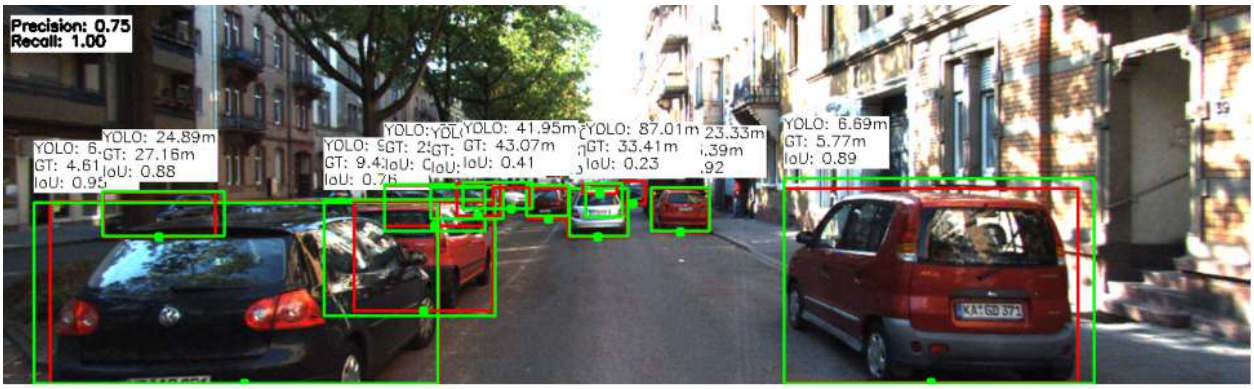

Figure 8: Frame Id: 006211

Figure 9: Frame Id: 006329

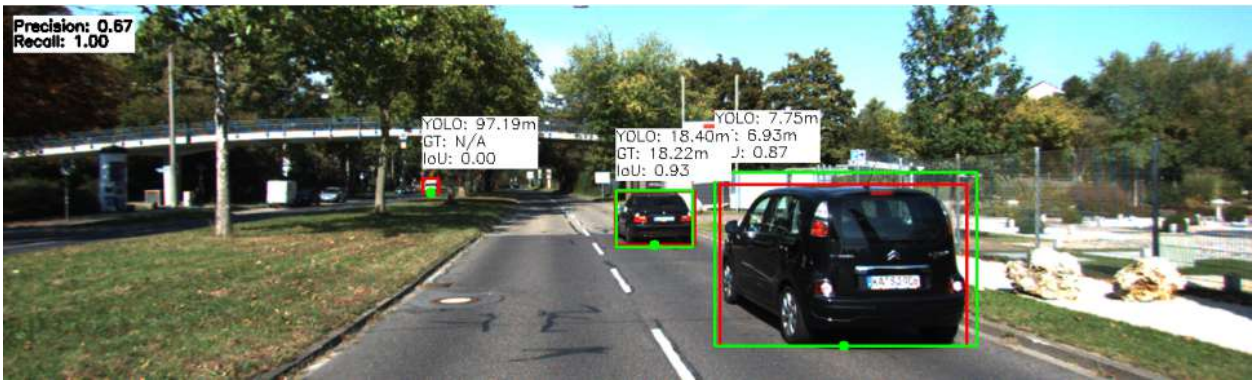

Figure 10: Frame Id: 006374

Figure 11: Frame Id: 006037

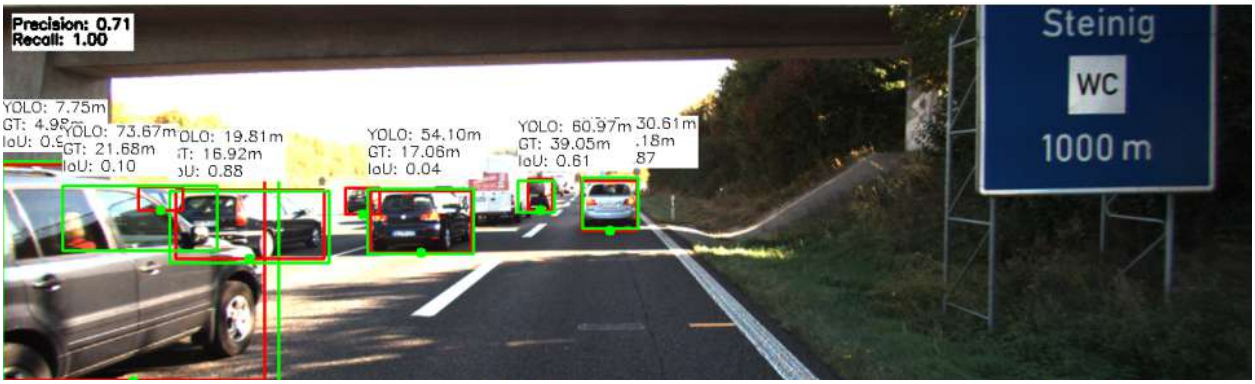

Figure 12: Frame Id: 006048

Figure 13: Frame Id: 006054

Figure 14: Frame Id: 006097

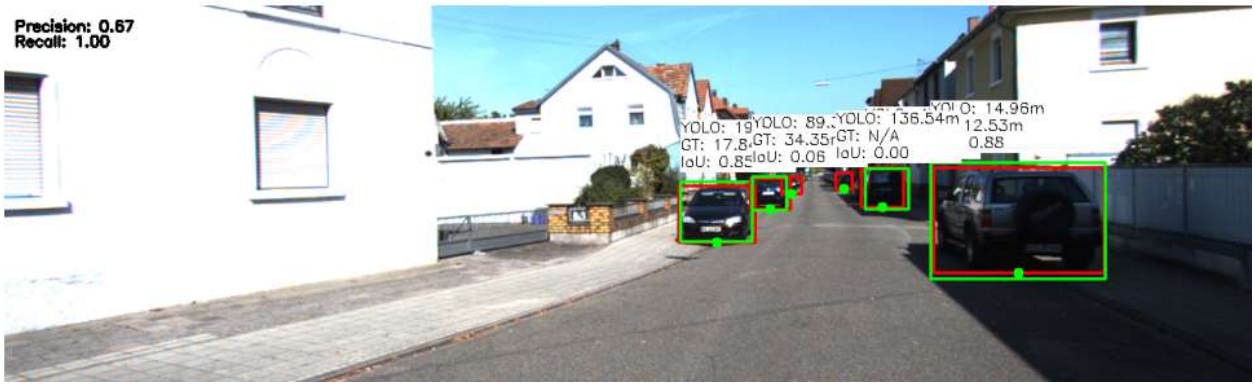

Figure 15: Frame Id: 006098

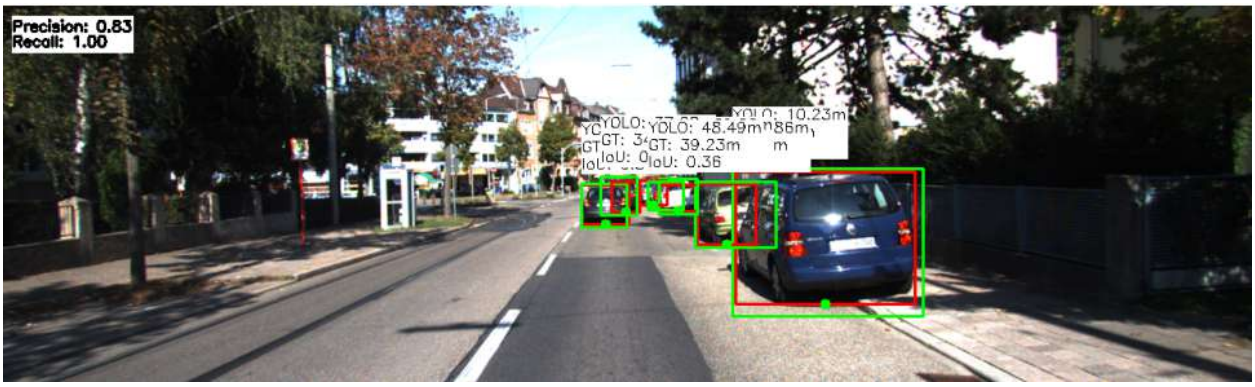

Figure 16: Frame Id: 006253

Figure 17: Frame Id: 006310

Figure 18: Frame Id: 006315

3.3 Images with no detection

The below images in which there is no detection occurred as there is no cars detected.

Figure 19: Frame Id: 006130

Figure 20: Frame Id: 006121

3.4 Overall result

The overall result can be shown with the help of plotting ground truth distance vs Yolo model distance on xy plane as shown below, As we can see the algorithm performs well up to a distance of around 40 meters,but then the deviation beyond this value can be seen increasing as object is getting far.

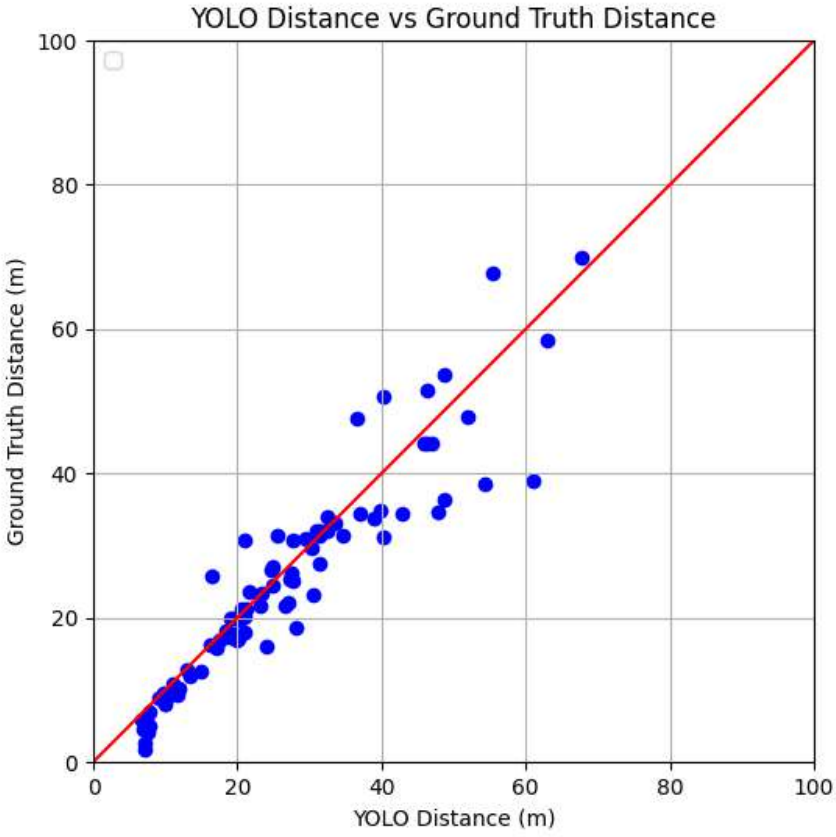

Figure 21: Plot between actual and estimated car depths

4 Conclusion

This study centered on object detection and depth estimation using the KITTI dataset. YOLOv8x was applied for object recognition, while intrinsic camera matrices were leveraged to calculate distances. Nonetheless, various challenges, such as occlusions, detection inaccuracies, uneven terrain, and fluctuating lighting conditions, hindered overall accuracy. Overcoming these hurdles requires the use of more advanced algorithms and precise calibration to enhance both accuracy and reliability. Our results highlight the importance of developing detection systems capable of managing real-world complexities. Future research should focus on improving calibration methods and creating adaptive algorithms to achieve more precise depth estimation in dynamic environments. These advancements are crucial for essential applications like autonomous vehicles and surveillance systems

References

[1] ELSER, Stefan. Computer vision. KITTI Selection. Available from: Dataset

[2] ELSER, Stefan. Computer vision. 3D-2D Projections Rotations. Available from: 3d-2d conversions