Enhancing the Accuracy and Speed of Object Detection and Distance Estimation to Improve the Safety of Autonomous Cars Movement

Faezeh Kabiri[1], Mahmoud Reza Delavar[2], Leila Hajibabi[3], Borzoo Nazari[4]

[1] GIS Department, School of Surveying and Geospatial Eng., College of Engineering, University of Tehran, Tehran, Iran, kabiri.faeze@ut.ac.ir
[2] Center of Excellence in Geomatic Eng. in Disaster Management and Land Administration in Smart City Lab., School of Surveying and Geospatial Eng., College of Engineering, University of Tehran, Tehran, Iran, mdelavar@ut.ac.ir
[3] Department of Industrial and Systems Engineering, North Carolina State University, US, lhajiba@ncsu.edu
[4] Geodesy Department, School of Surveying and Geospatial Eng., College of Engineering, University of Tehran, Tehran, Iran, borzoo.nazari@ut.ac.ir

Keywords: Object Detection, Distance Estimation, Autonomous Vehicles, Computer Vision, Spatial Data Quality

Abstract: Modern transportation systems face significant challenges in ensuring road safety, with approximately 35.1 million fatalities annually due to accidents, 93.5% of which are attributed to human errors. Autonomous vehicles have the potential to mitigate these incidents. They are classified into six levels of automation, from none to full automation, with obstacle detection and distance estimation being fundamental across all levels. This paper focuses on level 1 automation (driver assistance) in Iran, where most of the vehicles currently operate at Levels 0 and 1. It utilizes colour images captured by an SM-A52 mobile camera (2084 x 4624 pixels) in Districts 6 and 11 of Tehran under varying environmental and traffic conditions.
To enhance accuracy and speed in obstacle detection, four YOLO algorithm versions were implemented, with YOLOv8-s selected for its superior performance based on mean average precision, recall, and processing speed. For distance estimation, stereo imaging with two mobile cameras placed one meter apart was employed. Calibration parameters were obtained, and a 3D model was generated using Structure from Motion to calculate distances. The results were evaluated using Mean Absolute Error and Root Mean Squared Error, achieving a 20% increase in accuracy for obstacle detection compared to previous studies. Despite using more limited equipment, this research achieved comparable accuracy with respect to earlier works.

1. Introduction

In recent years, the increase in the number of vehicles and the desire for mobility in various locations has created a need for innovative methods to enhance safety and reduce accidents. The advancement of science and technology is shaping the concept of smart systems that interact with humans. By integrating geo sensor networks (GSN), artificial intelligence (AI) and information and communication technology (ICT), the idea of intelligent transportation system (ITS) emerges as a vital component of smart cities. In this context, autonomous vehicles play a crucial role, significantly improving the transportation experience through the integration of technology and transport.
According to the standard definitions by the Society of Automotive Engineers (SAE[1]), autonomous vehicles are categorized into six levels, ranging from no automation to full automation (Figure 1).

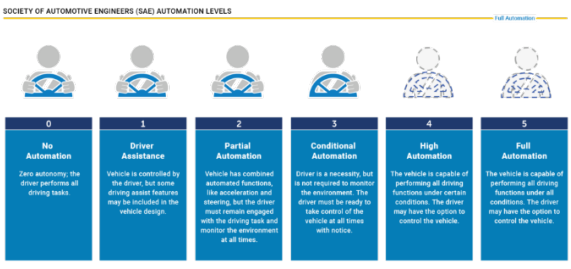


Figure 1. SAE Levels of driving automation (Musk., 2022)

Autonomous vehicles use various sensors and algorithms to perceive their environment, make decisions, and execute those decisions on the road. These vehicles primarily rely on the following systems (Betz et al., 2019) (Figure 2):

- **Sensors**: Devices such as cameras, Radar[2], and Lidar[3] provide raw data about the vehicle surroundings.
- **Perception**: Algorithms process sensor data to identify relevant objects (like other vehicles, pedestrians, or cyclists) and features (such as lanes or traffic signs).
- **Planning**: Based on the perceived environment and the vehicle current status, algorithms determine the actions the autonomous vehicle should take.
- **Control**: Algorithms convert the planned actions into commands that control the vehicle steering, acceleration, and Model Predictive Control (MPC).

This research aims to investigate and model a spatial approach to enhance the safety of autonomous vehicles. Safety in autonomous vehicles refers to the ability of the car to travel

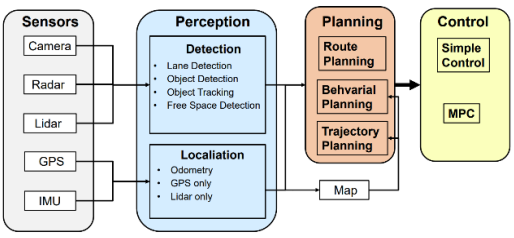


Figure 2. Software functions overview (Betz et al., 2019)

[1] Society of Automotive Engineers
[2] Radio Detection and Ranging
[3] Light Detection and Ranging

without accidents or collisions. To achieve this, the vehicle must perceive its surroundings and identify both static and dynamic obstacles. Environment perception and object identification are essential subsystems of autonomous vehicles, necessary for route planning, motion prediction, and collision avoidance (Qian et al., 2022) .Object identification is utilized at all levels of autonomous vehicles. As automation levels increase, more sensors are needed for better situation awareness. The proposed system employs a non-metric RGB camera as a sensor, and as most of Iranian vehicles fall within Level 0 or Level 1, Level 1 (driver assistance) is the practical application level considered in this research.

According to the World Health Organization (WHO), in 2018, approximately 1.35 million people died globally due to road traffic accidents, meaning one person is a victim every 24 seconds (Green, 2018). In Iran, this statistic was 17,716 fatalities in 2018 (WHO, 2018). These accidents have detrimental, psychological and economic impacts on society; thus, enhancing safety is a key component of intelligent transportation systems.

Accidents occur due to factors such as human errors, road conditions, vehicle issues, or environmental factors, with about 93.5% attributed to human errors (Winkle, 2016). The implementation of autonomous vehicles can reduce human errors, thereby enhancing safety, traffic control, comfort, convenience, travel time, energy efficiency, and environmental protection. A key requirement is the ability to accurately and rapidly identify and locate fixed obstacles (e.g., buildings, bridges, traffic signs) and moving obstacles (e.g., cars, pedestrians, motorcycles, bicycles) relative to the vehicle—taking into account safe distances, vehicle speed, acceleration, and the speeds of moving obstacles—to prevent collisions. Consequently, the research demonstrates that these methods increase vehicle safety, reduce accidents, bolster driver confidence, and improve overall driving experiences.

2. Literature Review

In the past two decades, it is widely accepted that the progress of object detection has generally gone through two historical periods including "traditional object detection period (before 2014)" and "deep learning based detection period (after 2014)" (Zou, 2019).

2.1 Traditional Detectors

Traditional object detection refers to the methods and techniques used before the advent of deep learning and neural networks (Zou, 2019).

2.2 Deep learning-based methods

Deep learning methods for object recognition are divided into two categories as follow (Qiao and Zulkernine, 2020):

2.2.1 Two-Stage Methods: The first category includes two-stage algorithms. In this category, the first stage detects likely regions in the image, and the second stage accurately identifies objects within those regions. Examples include Region-based Convolutional Neural Networks (R-CNN), Fast R-CNN, and Faster R-CNN (Zou, 2019).

2.2.2 One-Stage Methods: The second category consists of one-stage methods. In this category, probable regions and object identification are performed simultaneously. These methods detect objects through a single-stage analysis of the image. Examples include You Only Look Once (YOLO), Single Shot Detector (SSD), and Deconvolutional Single Shot Detector (DSSD) (Figure 3) (Zou et al., 2023).

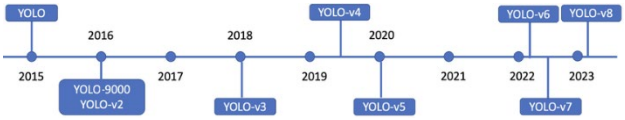


Figure 3. YOLO evolution timeline (Hussain, 2023)

Overview of Distance Estimation Methods

Various sensors such as stereo cameras, LiDAR, radar, monocular cameras, and combinations of these sensors are used for distance estimation (Ulusoy et al., 2023).

Stereo Vision

Various sensors (e.g., stereo cameras, LiDAR, radar, monocular cameras) are used for distance estimation (Ulusoy et al., 2023); in stereo vision, depth is obtained by computing disparity using methods such as Graph Cuts, Semi-Global Matching (SGM) (Zhou et al., 2020) and Block Matching (Fsian et al., 2022); (Irmisch, 2017). As photogrammetric techniques like Structure from Motion – Multi-View Stereo (SfM-MVS) (Smith et al., 2016); (Lowe, 2004) and feature extraction via Scale-Invariant Feature Transform (SIFT) enable cost-effective 3D reconstruction and robust object detection, this research demonstrates the successful implementation of these methods and the real-time results shown in Figure 4.

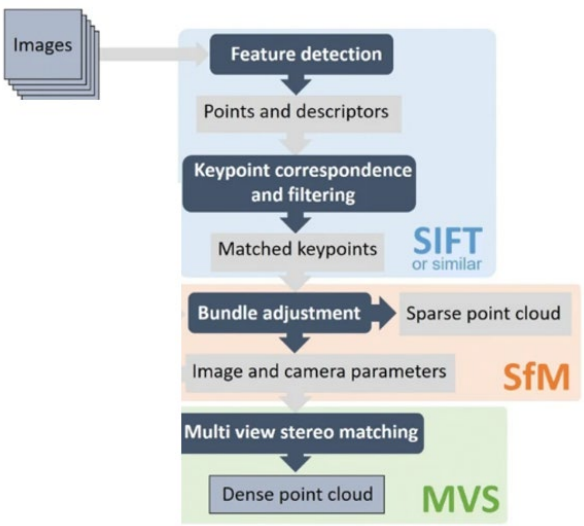


Figure 4. Schematic workflow of the SfM-MVS process resulting in a dense point cloud from image sets. The point cloud is georeferenced by providing positional information for images and/or ground control points (Iglhaut et al., 2019).

3. Material and Methods

3.1 Employed data

Image acquisition and labeling: This process involves collecting data at various times of the day in different parts of the study area and labeling it using the Roboflow system to prepare the data for

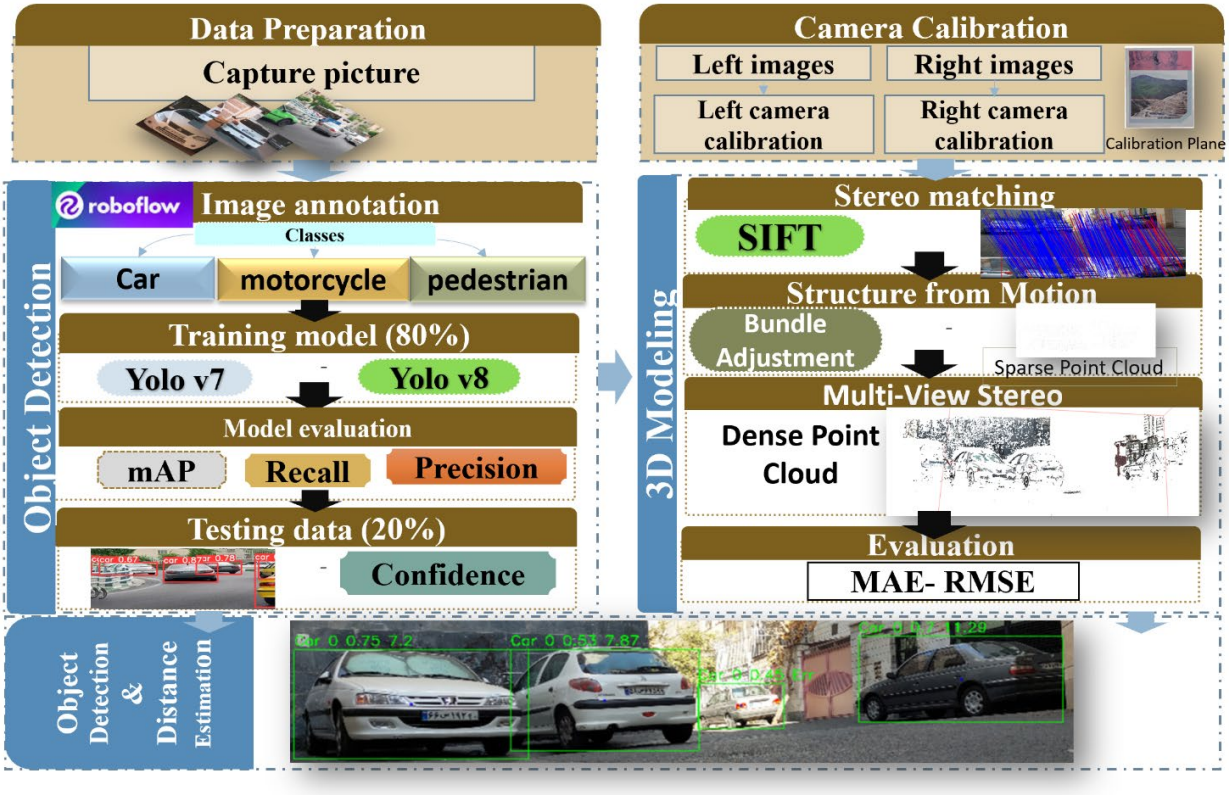


Figure 5. The process of object detection and distance estimation

training YOLOv7 and YOLOv8 for identifying cars, motorcycles, and pedestrians.
Training phase: Using 330 labeled images, the YOLOv7 and YOLOv8 models were trained. 80% of the data was used for training, 10% for validation, and 10% for testing the model (Liu et al., 2018).

Prominent feature detection images were captured using two identical mobile cameras positioned parallel to each other, one meter apart. The reason for selecting a one-meter distance is that, considering a greater distance between the two cameras, we achieve a stronger triangular geometry based on the angles formed in the triangle, resulting in higher accuracy. The closer the two cameras are, the weaker the geometry, which leads to lower accuracy in depth estimation, especially at greater distances. Furthermore, this distance was selected to ensure that distance estimation can be effectively performed in the overlapping region of the two images; as the distance between the two cameras increases, the overlapping section of the images decreases.

In the proposed research method to enhance safety in autonomous vehicles, advanced object detection and distance estimation techniques are integrated. The focal point of this study is to improve the accuracy of identifying cars, motorcycles, and pedestrians while introducing an innovative approach to estimate their distances. The goal is to create a safer driving environment for autonomous vehicles and to mitigate critical safety concerns.

3.2 Obstacle Detection

Obstacle detection in autonomous vehicles relies on camera sensor data processed by deep learning algorithms to identify obstacles such as cars, pedestrians, and motorcycles. The system outputs 2D coordinates, object class, and confidence scores, which are then used by vehicle control systems to prevent collisions.

3.2.1 Data Collection and Annotation: A fundamental step involved the collection of a dataset of images obtained using a mobile camera. This dataset was subsequently annotated with high accuracy for the classification and labelling of instances of cars, motorcycles, and pedestrians. The use of the Roboflow platform simplified the annotation process, which is a crucial step in obstacle detection—where higher accuracy in this task results in more precise outputs. The Roboflow platform enables us to perform annotations with high precision.
The goal of this research is the identification of cars, motorcycles, and pedestrians as significant dynamic and static obstacles for detection to support safe autonomous driving.
The methodology employed in this paper is illustrated in (Figure 5)

3.2.2 Obstacle Detection using YOLO algorithm: The YOLOv7 and YOLOv8 algorithms, known for their real-time object detection capabilities, were selected as the core method in this study. These algorithms were applied for the accurate identification and localization of target objects on the annotated images. The selection of YOLO algorithms is based on their proven performance in complex scenarios. The dynamics and movement in autonomous vehicles necessitate algorithms that can quickly and accurately detect objects. Therefore, YOLOv7 and YOLOv8, which are among the most up-to-date algorithms utilized in the field of object detection and boast the highest speed and accuracy compared to other object detection algorithms (Wen, 2019), were selected for use in this research. The following section will explain and compare YOLOv7 and YOLOv8 algorithms.

- YOLOv7: It outperforms all previously known object detectors in terms of both speed and accuracy, achieving a range of 5 frames per second to 160 frames

per second, with the highest accuracy of 56.8% AP[4] among all previously known online object detectors when using the COCO[5] dataset input at 30 frames per second or higher on a V100 GPU (Wang et al., 2022).

- YOLOv8: In January 2023, Ultralytics added YOLOv8 to the YOLO family. The comparisons indicate that YOLOv8 is recognized as the most advanced version of the YOLO. YOLOv8 is designed to be fast, accurate, and easy to use, making it an excellent choice for a wide range of object detection and tracking, instance segmentation, image classification, and pose estimation tasks (Hussain, 2023)(Figure. 6).

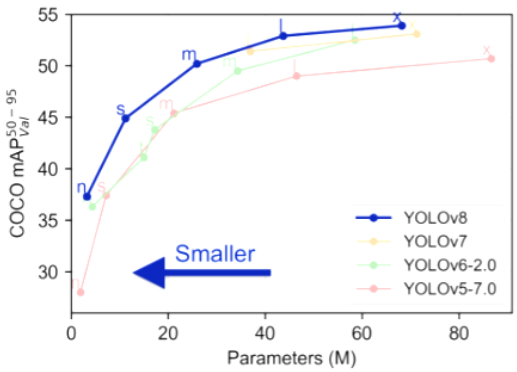


Figure 6. Performance chart of different YOLO versions based on the number of trained parameters (Jocher et al., 2023).

The following are some features of YOLOv8 compared to YOLOv7:

- Faster and More Accurate: According to Figure 6, YOLOv8 achieves a new high score of 53.7 mean Average Precision (mAP), marking a significant improvement over YOLOv7.
- Improved Model Architecture: YOLOv8 features enhanced architecture, including pose estimation models, which boost its object detection capabilities.
- Easier to Use: It offers a more user-friendly interface, simplifying the implementation and customization of various object detection tasks (Hussain, 2023).

3.3 Stereo Imaging and Distance Estimation

Using stereo imaging with two identical mobile cameras, this system calculates a depth map disparity to accurately estimate distances to obstacles—a crucial step for enhancing road safety in autonomous vehicles (Kok and Rajendran, 2019). The process involves the following six main steps:

3.3.1 Camera Calibration: Calibration parameters are obtained by imaging a plane with clearly defined control points using Agisoft software.

3.3.2 Image Loading and Correction: Stereo images from the left and right cameras are loaded and corrected using these calibration parameters.

3.3.3 3D Modelling Based on SfM-MVS: Using the SIFT algorithm for point matching, the relative positioning of the images is determined and 3D coordinates of feature points are acquired, resulting in a dense point cloud.

3.3.4 Obstacle Detection: The best-performing detection algorithm—balanced for speed and accuracy—is applied to the corrected image to identify obstacles by providing object class, bounding box, and confidence score.

3.3.5 Distance Calculation to Obstacles: The distance from the camera to each detected obstacle is computed using the depth map and the center of the bounding box.

3.3.6 Annotation: The image is annotated with bounding boxes and labels indicating object class, confidence score, and estimated distance, integrating object detection with distance estimation.

3.4 Evaluation of the Research Method

3.4.1 Performance Evaluation of Object Detection:

Model Training and Evaluation: Following the annotation process, the YOLOv7 and YOLOv8 models were precisely trained using the annotated dataset. The effectiveness of the models is measured through a systematic evaluation process. The following evaluation parameters are used to measure the effectiveness of object detection (Table 1):

Measure	Formula
Precision	TP / (TP + FP)
Recall (Sensitivity)	TP / (TP + FN)
F1 Score	2 * (Precision * Recall) / (Precision + Recall)
Mean Average Precision (mAP)	Calculated using precision-recall curves
Inference Speed	Frames per second (FPS) during real-time detection

Table 1. Evaluation measures. TP, TN, FP, P, N refer to the number of True Positive, True Negative, False Positive, Positive, and Negative samples, respectively

As shown in Table 1, a True Positive (TP) is a correct detection of an object that exists in the image. A False Positive (FP) is an incorrect detection of an object, A False Negative (FN) is an object that exists in the image but is not identified by the network. Finally, a True Negative (TN) is a correct detection of an object that does not exist in the image (Table 2).

	Predicted / Actual	
Negative	**Positive**	
(FP) False Negative	(TP) True Positive	**(P) Positive**
(TN) True Negative	(FN) False Positive	**(N) Negative**

Table 2. Confusion matrix

According to Bansal et al. (2020), accuracy is the percentage of correct predictions out of all predictions made on the dataset which indicates the correctness of classification. The detection accuracy is calculated based on Table 2 using Equation (3-1). The Equations (3-2), (3-3), (3-4), (3-5), and (3-6) are extracted from Park et al. (2021) and Bansal et al. (2020).

(3-1)

[4] Average Precision

[5] Common Objects in Context

$$\alpha = Correct\ prediction\ /\ Total\ prediction = (TP + TN)\ /\ (TP + TN + FP + FN) \quad (3\text{-}2)$$

$$Precision = TP\ /\ (TP + FP) \quad (3\text{-}3)$$

$$Recall = TP\ /\ (TP + FN) \quad (3\text{-}4)$$

$$F1 = 2 \times ((Precision \times Recall)/(Precision + Recall)) \quad (3\text{-}5)$$

$$AP = \sum_{n=1}^{N} (R_n - R_{n-1}) P_n$$

mAP is calculated based on the comparison between the predicted bounding box and the ground truth bounding box. For example, in Figure 7, the ground truth is represented by the red bounding box, while the detection is shown by the blue bounding box. The Intersection over Union (IoU) metric is used to calculate the mAP (Umar et al., 2022).

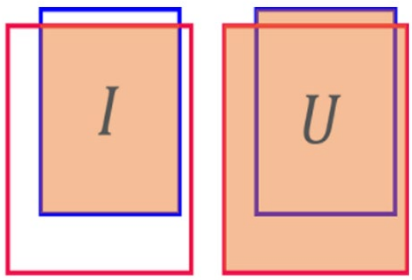


Figure 7. Intersection Over Union of the ground truth and predicted bounding boxes (Francies et al., 2022).

In Figure 7, the red box represents the ground truth, and the blue box represents the predicted bounding box (Francies et al., 2022).

$$IOU = \frac{Union}{Intersection} = \frac{U}{I} \qquad (0 \le IOU \le 1) \quad (3\text{-}6)$$

3.4.2 Evaluation of Distance Estimation Accuracy: To assess the accuracy of the distance estimation method, we utilized the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) as measurable metrices. These metrices serve as indicators of the precision of the method in estimating the distances of objects or obstacles. Low values of MAE and RMSE confirm the reliability and accuracy of the distance estimation process (Haris and Hou, 2020). These criteria demonstrate the accuracy of our predictions and indicate the level of deviation from actual values (Table 3).

Measure	Formula
Mean Absolute Error (MAE)	(1 / N) * Σ|estimated distance - ground truth distance|
Root Mean Square Error (RMSE)	√((1 / N) * Σ(estimated distance - ground truth distance)^2)

Table 3. Key formulas used to evaluate the accuracy of distance estimation methods (Gu, 2014)

3.4.3 Safety Improvement Evaluation: To evaluate safety improvement comprehensively, metrices such as collision detection rate, early warning rate, and false positive rate are used. These metrices assess the system ability to identify collision risks and provide timely warnings. Given our focus on the perception component of autonomous vehicles, and the fact that safety improvement metrices require both an implemented system and vehicle control system, calculating safety improvement metrices is beyond the scope of this study.

4. Results and Analysis

4.1 Study Area

The study area encompasses Azadi Street, Enghelab-e Eslami Street, and the Kurdistan Highway within Districts 6 and 11 of Municipality of Tehran, capital of Iran. Data were collected in both static and dynamic conditions at various times of day and night to examine the impact of daylight and nighttime lighting on object detection and distance estimation for the identified objects. The study area was selected to include a variety of traffic conditions and types of obstacles (Figure 8).

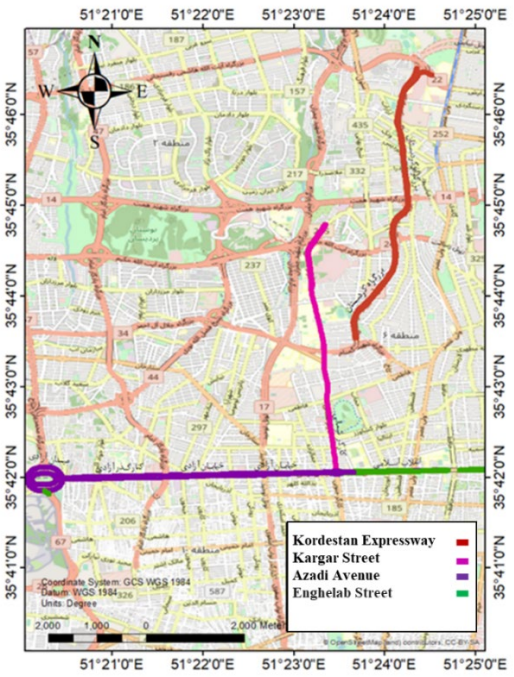


Figure 8. The study area

4.2 Training YOLOv7 and YOLOv8 Models

4.2.1 Obstacle Detection Model Training Process

Images in the training subset were resized to 640 × 640 pixels to meet the input requirements for the selected architecture. Multiple augmentations were applied to these images to enhance model performance during training. The object detection models were trained on a desktop computer with access to Google Colab, enabling computations on a Tesla K80 GPU with 15 GB of memory. In the initial phase, a dataset was collected, and various YOLO models were adapted to suit vehicle, motorcycle, and pedestrian detection tasks. The models were trained and validated until the loss function reached a stable state, with the mean losses

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-G-2025-435-2025 | © Author(s) 2025. CC BY 4.0 License.

fluctuating within an acceptable threshold. The model prediction confidence for the presence of potholes within each specified frame was recorded. Both YOLOv7 and YOLOv8 approaches were used for training.

4.2.2 Key Parameters for Model Training

Training required tuning of three parameters including batch size, number of epochs, and IoU threshold. Important considerations for them are outlined below:

Batch Size: A larger batch size can accelerate convergence during training, although it requires more memory and computational resources. Batch size selection depends on available hardware and dataset size (Sirisha et al., 2023). Given the available hardware and the number of input images, a batch size of 32 was used.

Number of Epochs: The number of epochs determines how many times the model iterates over the entire dataset during training. Higher epoch counts allow the model to learn more patterns, potentially improving performance, however, excessively high values may lead to overfitting. The optimum number of epochs depends on the dataset complexity and model convergence (Sirisha et al., 2023). Through the experiments and their evaluation on validation data, 55 epochs were selected for training the YOLO algorithms.

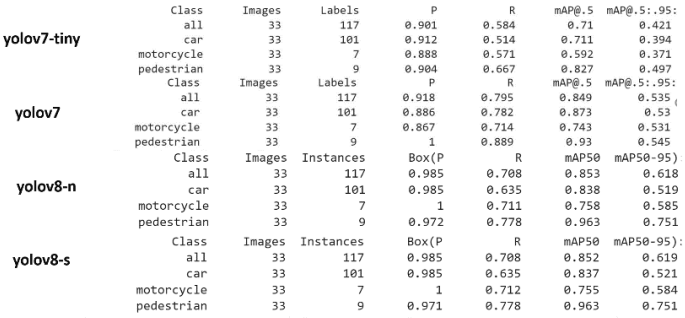
IoU Threshold: The IoU threshold, as an input parameter, specifies that objects identified in the image are considered obstacles if their IoU exceeds this threshold. Based on prior studies (Arif et al., 2023) and the F1-curve, this threshold was set at 50% for this research. Optimization methods can be employed to determine the optimum threshold, which can be further explored in future research.

4.2.3 Training the YOLOv7 Model: In this study, two models including YOLOv7 and YOLOv7-tiny, were used for training. YOLOv7 is a more accurate and powerful object detection model, while YOLOv7-tiny is a smaller, lightweight version optimized for edge GPU computations.

4.2.4 Training the YOLOv8 Model: YOLOv8 is available in various sizes including Yolov8n[6], Yolov8s[7], Yolov8m[8] , Yolov8l[9], and Yolov8x[10]. In this paper, the YOLOv8-s and YOLOv8-n models were used for training.

4.2.5 Evaluation of the Obstacle Detection Models: In the evaluation phase of YOLO-based object detection algorithms, a set of metrices and various charts are utilized to comprehensively assess the performance of these algorithms. These metrices include bounding boxes for detected objects, object confidence scores, classification accuracy, precision, recall, bounding boxes in the validation set, object confidence in the validation set, mean Average Precision (mAP) at a 0.5 threshold, and mAP at a 0.95 threshold. In addition, metrices such as the F1-curve, recall curve, precision curve, and precision-recall curve are employed (Padilla et al., 2020).

These evaluations enable a detailed analysis of the performance of YOLO algorithms in detecting and classifying objects in images, allowing for the identification of areas where improvement is required. By analyzing these metrices and observing the curves, it can be ensured that the algorithms achieve a balance between precision and recall, optimizing object detection accuracy (Figure 9 and Table 4).



	Class	Images	Labels	P	R	mAP@.5	mAP@.5:.95:
yolov7-tiny	all	33	117	0.901	0.584	0.71	0.421
	car	33	101	0.912	0.514	0.711	0.394
	motorcycle	33	7	0.888	0.571	0.592	0.371
	pedestrian	33	9	0.904	0.667	0.827	0.497
	Class	Images	Labels	P	R	mAP@.5	mAP@.5:.95:
yolov7	all	33	117	0.918	0.795	0.849	0.535
	car	33	101	0.886	0.782	0.873	0.53
	motorcycle	33	7	0.867	0.714	0.743	0.531
	pedestrian	33	9	1	0.889	0.93	0.545
	Class	Images	Instances	Box(P	R	mAP50	mAP50-95)
yolov8-n	all	33	117	0.985	0.708	0.853	0.618
	car	33	101	0.985	0.635	0.838	0.519
	motorcycle	33	7	1	0.711	0.758	0.585
	pedestrian	33	9	0.972	0.778	0.963	0.751
	Class	Images	Instances	Box(P	R	mAP50	mAP50-95):
yolov8-s	all	33	117	0.985	0.708	0.852	0.619
	car	33	101	0.985	0.635	0.837	0.521
	motorcycle	33	7	1	0.712	0.755	0.584
	pedestrian	33	9	0.971	0.778	0.963	0.751

Figure 9. Model performance output on 33 test images

	Precesion (%)	Recall (%)	mAP 0.5 (%)	mAP 0.95 (%)	Speed (%)
YOLOv7	91	80	85	53	47
YOLOv7-tiny	90	59	71	42	140
YOLOv8-n	90	70	85	61	61
YOLOv8-s	90	70	85	62	61

Table 4. The performance of different models for all classes

4.3 Distance Estimation to Identified Obstacles Data Acquisition:

Stereo images were captured by positioning two identical mobile cameras parallel to each other with a distance of one meter apart. As the distance between the two cameras increases, the overlap decreases, resulting in depth estimation being limited to a smaller area, as it is only possible within the overlap region. In addition, depth estimation is more effective at shorter distances from the vehicle when the cameras are closer together. However, with greater distances, the robustness of triangulation increases, and the computed extrinsic parameters are more accurate, leading to more precise distance calculations for obstacles from the vehicle. Therefore, in this study, considering the importance of distance

[6] Yolov8 nano
[7] Yolov8 small
[8] Yolov8 medium
[9] Yolov8 large
[10] Yolov8 x-large

estimation for obstacles closer to the vehicle, a high accuracy in distance estimation, the width of the vehicle, the area of the overlap region covered by the captured images, and a one-meter separation between the two cameras were selected (Figure 10).

Figure 10. Camera Setup and Imaging Geometry.

4.3.1 Camera Calibration: Using Agisoft software, camera calibration parameters were estimated, including detailed camera specifications, calibration coefficients, and the corresponding correlation matrix.

4.3.2 3D Modeling and Distance Estimation: To estimate the distance to obstacles, the SfM-MVS structure was utilized. After generating the point cloud and obtaining the camera coordinates and the 3D coordinates of points in the model space, the distances from the camera to the obstacles were calculated. To obtain the 3D coordinates of points, it is necessary to perform the matching process using the SIFT algorithm, relative orientation of images through ray bundle equations, and the creation of a dense point cloud (Figure 11).

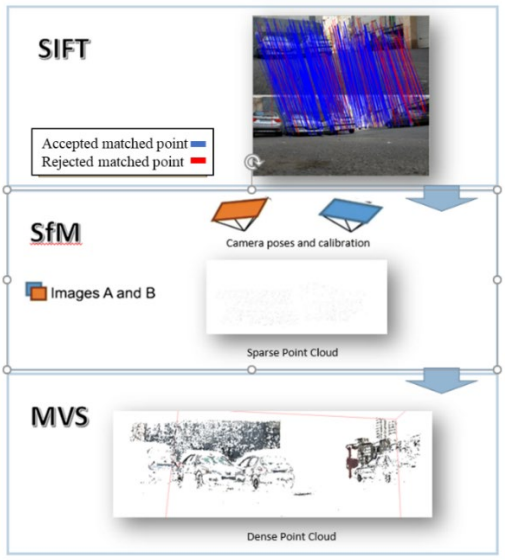


Figure 11. Three key stages in an SfM-MVS workflow:(1) Key point detection and matching (e.g., SIFT), (2) SfM with camera parameters and a sparse point cloud as output, (3) Dense point cloud generation after Multi-View Stereo (MVS).

4.4 Evaluation Stage and Performance Metrices

Comparison chart of actual versus predicted distances is presented in Figure 12.

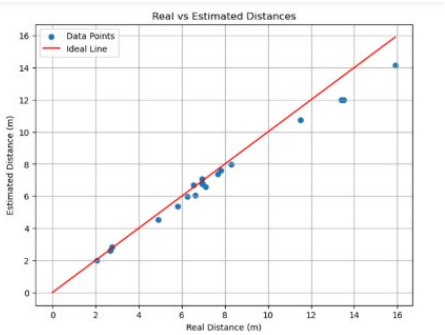


Figure 12. Comparison chart of actual versus predicted distances.

The performance evaluation of distance estimation to the identified obstacles is calculated based on the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) metrices for distance estimation as presented in Table 5.

MAE (m)	0.48
RMSE (m)	0.69

Table 5. The Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) metrices for distance

4-5 Calculation of Distance to The Detected Objects

In this study, the calculated distances to the centers of identified obstacles were compared with actual values using Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) metrics. Due to equipment limitations, distances were estimated using fixed cameras relative to stationary objects, considering potential inaccuracies.

Distance calculation was performed using Structure from Motion (SfM) and stereo vision. Two cameras captured images from different perspectives and a 3D model was generated. The SIFT algorithm was used for feature matching, and depth information was extracted by computing disparities between corresponding points in both images.

However, this process is not real-time due to the following limitations:

- High computational cost of SfM, which results to extensive feature processing.
- Limited processing power of GPUs and CPUs, leading to slow execution.
- Large data volume in stereo vision, demanding significant memory and bandwidth.

Future research can optimize algorithms and leverage advanced hardware (e.g., high-performance GPUs and neural networks) to enable real-time processing. Despite current limitations, this method offers a cost-effective alternative to LiDAR-based systems. The final results of obstacle detection and distance estimation are illustrated in Figure 13.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-G-2025-435-2025 | © Author(s) 2025. CC BY 4.0 License.

Figure 13. The image captured by the left camera includes annotations for each class, indicating the confidence percentage and the distance to each detected object.

5. Discussion

This study compared the performance of YOLOv8 for obstacle detection and stereo vision-based distance estimation with previous works. Below, we present a detailed comparison with prior research, using the data from Tables 6 and 7 to assess how our results align with existing methods.

Speed (FPS)	mAP %	Classes	Model	Source
45	69	People-car-trunk-bus- motorcycle-bicycle	YOLO	Aryal, (2018)
17	65.96	Person- car- traffic light- motorcycle	YOLOv4	Gao et al, (2021)
67	85	Person- car- motorcycle	YOLOv8	This research

Table 6. Obstacle detection comparison

As shown in Table 6, the YOLOv8 algorithm in this study achieved an accuracy of 85%, outperforming Aryal (2018) with 69% accuracy. This improvement highlights the enhanced detection capabilities of YOLOv8 compared to earlier YOLO versions.

Similarly, for distance estimation, our approach achieved a Mean Absolute Error (MAE) of 0.48 meters, which is comparable to previous studies that used more advanced sensor setups (see Table 7)

Precision for less than 30 m (%)	Sensor	Source
95.58	Camera	Kumar et al.(2020)
98.68	Lidar	
98.02	LiDAR &Camera	
94.88	Camera	This research

Table 7: Distance estimation comparison

6. Conclusion

The inception of autonomous vehicles offers a wide range of benefits, including enhanced safety, improved traffic management, increased comfort, reduced travel times, energy savings, and environmental sustainability. Autonomous vehicles are categorized into six levels of automation, with higher levels representing more advanced capabilities toward full autonomy. In Iran, most of the vehicles are currently operating at Level 0 and Level 1 automation. This research focuses on obstacle detection and distance estimation at Level 1 (driver assistance) using camera sensors.

This study investigated obstacle detection and distance estimation in autonomous vehicles, emphasizing the importance of accuracy and speed in detection algorithms. The YOLOv8 algorithm was employed to identify cars, motorcycles, and pedestrians with high precision and efficient processing speed. In addition, distance estimation was achieved through stereoscopic vision, utilizing two cameras placed at a fixed distance in controlled conditions. To improve the precision of distance estimations, a Structure from Motion (SfM) framework for 3D modeling was integrated, along with the SIFT algorithm for feature matching.

It is also worth noting that while this research used YOLOv8, the latest version of YOLO, v11, has recently been released, offering potential improvements in detection capabilities and performance.

References

Arif, M. ul I., Jameel, M., Schmidt-Thieme, L., 2023: Directly Optimizing IoU for Bounding Box Localization. arXiv preprint, arXiv:2304.07256. https://doi.org/10.48550/arXiv.2304.07256.

Bansal, K., Mittal, K., Ahuja, G., Singh, A., Gill, S.S., 2020. DeepBus: Machine learning based real time pothole detection system for smart transportation using IoT. Internet Technology Letters 3, e156.

Betz, J., Wischnewski, A., Heilmeier, A., Nobis, F., Stahl, T., Hermansdorfer, L., Lienkamp, M., 2019. A Software Architecture for an Autonomous Racecar. https://doi.org/10.1109/VTCSpring.2019.8746367

Francies, M.L., Ata, M.M., Mohamed, M.A., 2022. A robust multiclass 3D object recognition based on modern YOLO deep learning algorithms. Concurrency and Computation: Practice and Experience 34, e6517.

Fsian, H., Mohammadi, V., Gouton, P., Minaei, S., 2022. Comparison of Stereo Matching Algorithms for the Development of Disparity Map. https://doi.org/10.48550/arXiv.2210.15926

WHO | By category | Road traffic deaths - Data by country .,2018. WHO. URL https://apps.who.int/gho/data/view.main.51310?lang=en (accessed 7.14.23).

Green, B., 2018. World Health Organisation (WHO) releases the Global Status Report on Road Safety 2018. iRAP. URL https://irap.org/2018/12/world-health-organisation-who-releases-the-global-status-report-on-road-safety-2018/ (accessed 7.14.23).

Gu, T., 2014: Real-Time Obstacle Depth Perception Using Stereo Vision. *Master's Thesis*, University of Florida, US.

Haris, M., Hou, J., 2020. Obstacle Detection and Safely Navigate the Autonomous Vehicle from Unexpected Obstacles on the Driving Lane. Sensors (Basel, Switzerland) 20. https://doi.org/10.3390/s20174719

Hussain, M., 2023. YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection. Machines 11, 677. https://doi.org/10.3390/machines11070677

Iglhaut, J., Cabo, C., Puliti, S., Piermattei, L., O'Connor, J., Rosette, J., 2019. Structure from Motion Photogrammetry in Forestry: a Review. Current Forestry Reports 5. https://doi.org/10.1007/s40725-019-00094-3

Umar, A., Zamzuri, H., Limbu, D.K., 2022: Internet of Vehicle (IoV) Applications in Expediting the Implementation of Smart Highway of Autonomous Vehicle: A Survey. *Springer Professional*. https://www.springerprofessional.de/en/internet-of-vehicle-iov-applications-in-expediting-the-implement/16065242 (accessed 17 Aug 2023).

Irmisch, P., 2017: Camera-Based Distance Estimation for Autonomous Vehicles. *Master’s Thesis*, Technische Universität Berlin, Faculty of Electrical Engineering and Computer Science, Department of Computer Vision and Remote Sensing, Berling, Germany.

Jocher, G., Chaurasia, A., Qiu, J., 2023. YOLO by Ultralytics.

Liu, C., Tao, Y., Liang, J., Li, K., Chen, Y. (2020). Object Detection Based on YOLO Network. In: Proceedings of the IEEE 4th Information Technology and Mechatronics Engineering Conference (ITOEC), 26–28 February 2020, Coimbatore, India, pp. 799–803. IEEE. https://doi.org/10.1109/ICICT48043.2020.9112424

Lowe, D.G., 2004: Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.*, **60**, 91–110. https://doi.org/10.1023/B:VISI.0000029664.99615.94.

Musk, E., 2022. SAE levels of automation in cars simply explained. Electic Spare Parts. URL https://electicspareparts.com/sae-levels-of-automation-in-cars-simply-explained/ (accessed 3.7.25).

Padilla, R., Netto, S., da Silva, E., 2020: A Survey on Performance Metrics for Object-Detection Algorithms. IEEE IWSSIP, https://doi.org/10.1109/IWSSIP48289.2020.

Park, S.-S., Tran, V.-T., Lee, D.-E., 2021: Application of Various YOLO Models for Computer Vision-Based Real-Time Pothole Detection. Appl. Sci., 11, 11229.

Qian, R., Lai, X., Li, X., 2022: 3D Object Detection for Autonomous Driving: A Survey. Pattern Recognit., 130, 108796. https://doi.org/10.1016/j.patcog.2022.108796.

Qiao, D., Zulkernine, F., 2020: Vision-Based Vehicle Detection and Distance Estimation. 2020 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 2836–2842. https://doi.org/10.1109/SSCI47803.2020.9308364.

Sirisha, U., Praveen, S.P., Srinivasu, P.N., Barsocchi, P., Bhoi, A.K., 2023: Statistical Analysis of Design Aspects of Various YOLO-Based Deep Learning Models for Object Detection. Int. J. Comput. Intell. Syst., 16, 126. https://doi.org/10.1007/s44196-023-00302-w.

Smith, M.W., Carrivick, J.L., Quincey, D.J., 2016: Structure from Motion Photogrammetry in Physical Geography. Prog. Phys. Geogr. Earth Environ., 40, 247–275. https://doi.org/10.1177/0309133315615805.

Ulusoy, U., Eren, O., Demirhan, A., 2023: Development of an Obstacle Avoiding Autonomous Vehicle by Using Stereo Depth Estimation and Artificial Intelligence-Based Semantic Segmentation. Eng. Appl. Artif. Intell., 126, 106808. https://doi.org/10.1016/j.engappai.2023.106808.

Wang, C.-Y., Bochkovskiy, A., Liao, H.-Y.M., 2022: YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. arXiv preprint, arXiv:2207.02696. https://doi.org/10.48550/arXiv.2207.02696.

Wen, S., 2019: Object Detection with YOLO | Bringing Vision to Self-Driving Cars [WWW Document]. Medium. URL: https://towardsdatascience.com/object-detection-with-yolo-bringing-vision-to-self-driving-cars-980295226830 (accessed 17 Aug 2023).

Winkle, T., 2016: Safety Benefits of Automated Vehicles: Extended Findings from Accident Research for Development, Validation, and Testing. In: Maurer, M., Gerdes, J.C., Lenz, B., Winner, H. (Eds.), Autonomous Driving: Technical, Legal and Social Aspects, Springer, Berlin, Heidelberg, pp. 335–364. https://doi.org/10.1007/978-3-662-48847-8_17.Zhou, K., Meng, X., Cheng, B., 2020: Review of Stereo Matching Algorithms Based on Deep Learning. Comput. Intell. Neurosci., 2020, 8562323. https://doi.org/10.1155/2020/8562323.

Zou, X. (2019). A Review of Object Detection Techniques. In: Proceedings of the 2019 International Conference on Smart Grid and Electrical Automation (ICSGEA), 10–11 August 2019, Xiangtan, China, pp. 251–254. IEEE. https://doi.org/10.1109/ICSGEA.2019.00065

Zou, Z., Chen, K., Shi, Z., Guo, Y., Ye, J., 2023: Object Detection in 20 Years: A Survey. arXiv preprint, arXiv:1905.05055. https://doi.org/10.48550/arXiv.1905.05055.