A NOBEL FRAMEWORK OBSTACLE DETECTION FOR UAV & DRONE USING YOLO V8 AND HC-SR04 ULTRASONIC SENSOR

Gourab Biswas
School of Computer Engineering
KIIT Deemed to be University
Bhubaneswar,India
gourabiswas2003@gmail.com

Abstract:

The current paper dives deep into the complications and inspects the development of a real-time obstacle detection system designed for UAVs and drones with the integration of advanced computer vision techniques and ultrasonic sensor technology. Here YOLO (You Only Look Once) version 8 is implemented in Python 3.11 which detects, locates and segments obstacles more accurately. This combination of object detection, location, bounding box, and motion analysis conveys the exact state and shape of the existing object in the environment so that the drone or the remote pilot could locate and differentiate between stationary and moving objects. This will be more user friendly as it will be able to detect and track the moving obstacles, including pedestrians, vehicles, birds or other dynamic elements as well as the static ones. The YOLOv8 model is chosen due to its high precision for detection with added segmentation masks that outline the shape of obstacles, giving better situational awareness. By introducing the HCSR-04 ultrasonic sensor also complements this by measuring distance from the drone to detected obstacles. The HCSR-04 works on the principle of transmitting ultrasonic waves and then later using the time that reflected waves take to come back (SONAR) the sensor measures the accurate distance in real time. It uses an Arduino Uno micro controller so that it can control and process the outputs of the HCSR-04 in order to amplify the flexibility and responsiveness of the system. The integration of YOLOv8 visual data and ultrasonic sensor distance measurements results in as a fully assistive obstacle detection tool. Real-time operation becomes pertinent for dynamic environments where obstacles could appear and disappear shortly. The system creates the opportunity for immediate responses on safety concerns for the drone. The use of a cost-effective design approach has been applied to make this kind of high-end technology related to obstacles detection available to a larger segment of autonomous drones. The proposed system demonstrates experiments which assures that it works perfectly in different circumstances, accurately segmenting and localizing obstacles. The autonomous UAVs or drones get feedback in correct time which help them avoid obstacles and make their way through safely in complex surroundings by using the high-performance, user-friendly and affordable solution of integrated object detection capability of YOLOv8 with Arduino-controlled ultrasonic sensing.

Key Words: YOLOv8, HC-SR04, Ultrasonic Sensor, Computer Vision, Image Segmentation, Sensor Fusion.

Introduction:

Drones which are capable of flying at a low height are highly essential now a days due to their wide applications in Surveillance, Security, Emergency Response, logistics, firefighting, Construction sites, Precision agriculture, Traffic monitoring, Crowd monitoring etc. The most important benefit is that there is no life risk. These autonomous systems facilitate safe and secure inspections in complicated and dangerous environments. Navigational hindrances are the most rigorous problem that autonomous robots face. In congested surroundings they face obvious obstacles. During a significant amount of time the drone has being controlled and driven by the remote pilots which undoubtedly commenced the first steps of this technological experiment but as the developments are taking place the charges are becoming expensive and difficult to recruit efficient and trained professionals. Moreover, there always remains an unavoidable chance for human error which can cause serious accidents and monetary loss. In these circumstances the technological solution to build autonomous drones which would work like the ultimate savior that promises to detect real-time obstacles in the environment. Therefore, the autonomous drone can lead an independent and hurdle free flight.

AI, ML integrated Computer vision and sensor technologies have recently made tremendous advancement and present promising alternatives to previous human aid. Technological tools are combined in a way that even the distanced obstacles can be located and can be differentiated between stationary and moving objects. Among these object detection models, YOLO (You Only Look Once) gained significant popularity because it provides fast and accurate object detection in real time. YOLOv8 is the latest version (2023), which enhances the success of earlier iterations by increasing speed and accuracy. This work proposed an idea to develop a new system combining the detection, location and segmentation of objects with movement analysis utilizing the YOLOv8 model. Over that HCSR-04 ultrasonic sensor is used to measure the distance of obstacles detected. YOLOv8 produces object detection. Along with that image segmentation is incorporated to provide a clear definition of the obstacles ahead with x and y coordinates. The system also clearly distinguishes between static and dynamic objects which help the drone to safely navigate through complex environments.

This computer vision technique relies on 2D images. From that only X and Y coordinate of the obstacle can be identified but the Z coordinate means the distance of the obstacle from the drone cannot be calculated. To measure the distance between the drone and detected obstacles, an ultrasonic sensor, HCSR-04, is added. Measuring such distance will be important to establish whether the obstacle is within a critical range that may require immediate action. While YOLOv8 determines and distinguishes a static or dynamic obstacle with their X & Y coordinates, the ultrasonic sensor gives feedback of Z coordinate so that the system can get an approximation of how close the drones are to it. This sensor fusion approach brings the system to deliver not only spatial awareness but also depth perception to improve the safe navigation capabilities.

Additionally, its real-time processing capabilities ensure immediate feedback, an essential requirement for assistive technologies aimed towards autonomous navigation. The rapid and efficient detection provided by YOLOv8 and integrated image segmentation feature ensures that the information relating to obstacles reached by the system is very timely; continuous measurement of distance using the ultrasonic sensor also offers a reliable and responsive way of assessing proximity to such obstacles. Taking together, these technologies provide all-around situational awareness, alerting the system to potential hazards and giving the drone ample time to make adjustment in movement or direction. All this was to be done affordably since the components that are going to be used are off-the-shelf, easier to integrate, and cost efficient. The HCSR-04 ultrasonic sensor is well known because of its great accuracy and low cost; hence, it is a good sensor for real-time distance measurement in assistive devices. It combines with the computational power of YOLOv8 to form a very accessible solution for obstacle detection.

Literature Review:

Ahmed Ben Atitallah et al proposed a work in which they created a navigation system for visually impaired and blind persons using YOLO V5. They have trained the model using 2 datasets; one for indoor condition and another for outdoor condition and implemented on a Xilinx ZCU 102 board [1].

Marta Lalak and Damian Wierzbicki used the YOLO algorithm to detect obstacles for UAVs. They have located the X&Y coordinate of the centroid of the obstacle [2].

H. Y. Lee et al proposed an idea of obstacle detection in UAV technology using Faster Region-based Convolutional Neural Network Approach. They have trained their model using COCO dataset and used bounding box for detecting the obstacle [3].

R. VAIRAVAN, S. AJITH KUMAR et al came up with an idea to detect obstacles for robotic vehicles using ultrasonic sensors which were incorporated with an Arduino micro controller. The programming was done in an android app [4].

Sanjukumar NT and P Rajalakshmi came up with an idea to detect obstacle and avoid collision using rangefinder sensors like ultrasonic sensor, lidar light v3 sensor with kalman filter [5]

Lingyun Shen; Baihe Lang; Zhengxun Song proposed a paper that presents a novel approach to object detection in complex remote sensing images using the DS-YOLOv8 model. Enhanced by the Deformable Convolution C2f (DCN C2f) and Self-Calibrating Shuffle Attention (SC SA) modules, the model improves receptive field adjustment and multi-scale feature learning, addressing challenges in detecting multi-scale, occluded, and small objects [6].

Suherman Suherman; Rizky Ananda Putra; Maksum Pinem proposed a system where six ultrasonic sensors along with arduino were used to detect obstacle for a quadcopter-based drone[9].

Researchers have been using many object detection architectures for obstacle avoidance in many fields. Some of the renowned frame works are YOLOv8, Mask R-CNN, Faster R-CNN, RetinaNet and SSD. All have some pros and flaws which are discussed in Table: 1.

Table 1: Comparison of largely used object detection architectures

Feature	SSD	Mask R-CNN	Faster R-CNN	RetinaNet	Proposed work Using (YOLOv8)

Speed	Very fast	Low	Moderate	Fast	Very fast
Accuracy	High	Very High (for segmentation)	High	High	High
Segmentation	No	Yes	No	No	No
Real-time performance	Good	Low (not suited for real time)	Moderate (slow for real time)	Good	Excellent
Strengths	Fast, simple architecture	High precision, instance segmentation	Robust detection, high accuracy	Handles classes imbalance well	Fast, high detection accuracy
Weaknesses	Lower accuracy, struggles with small objects	Slow, high computational cost	Slow limited real time performanc e	Lower precision than two stage models	No segmentation, limited flexibility

You Only Look Once (YOLO) is a renowned one stage detector architecture. It is an extremely fast architecture network inspired by GoogleNet, commonly known as Darknet. The network consists of 24 convolutional layers which work to extract feature and 2 fully connected layers for the predictions in which the framework is trained on the ImageNet-1000 dataset. The architecture is shown in Figure 1 [7]. It uses algorithms which are based on regression where the process of detection, localization, and classification of the object for the input image will take place in a single pass. The detection process starts with re-sizing the input image into 448 X 448 and then divided into S X S grid cell, then fed into the single convolutional network. Each grid cell predicts bounding box and confidence score which output of each bounding box consists of 5 prediction values: offset values (x, y, w, and h) where x and y represent the object's coordinates in the input image and w and h stand for the object's width and height, respectively, for each bounding box, while the last prediction value is confidence score or class probabilities that given in terms of an IOU (intersection over union), which should have the object exist in the bounding box. Location the item in the image is done by selecting the bounding box with a high confidence score over the threshold value.

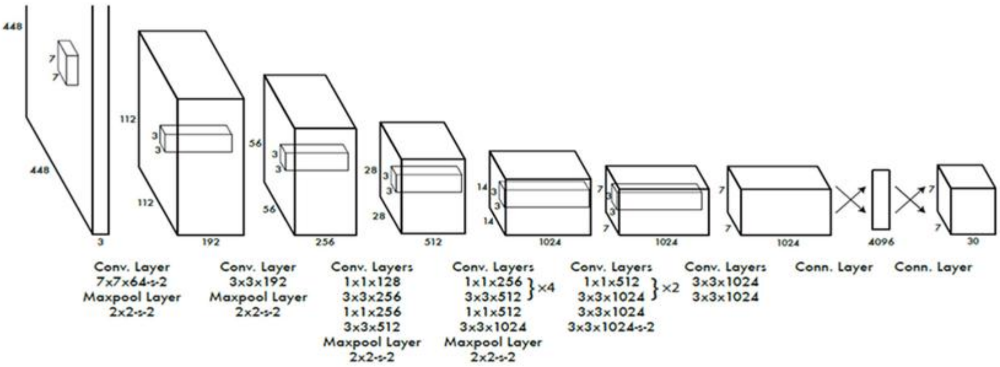

Figure: 1. YOLO architecture [7].

There are many variants of YOLO currently available. YOLOv1 was introduced in 2016, and the latest one is YOLOv8 which has been brought out in 2023. In total there are 8 variants. Gradually the processing speed and accuracy is increased, and the backbone architecture is altered as per requirements. The latest YOLOv8 is based on transformers supported CSPDarknet53, ResNet, EfficientNet backbone architecture which is very efficient in its work. Along with speed and accuracy the transformer support and anchor-free technology is the key improvement of YOLOv8 over the previous version [7].

The HC-SR04 ultrasonic sensor is a low-cost sensor by which the distance of any object from the sensor can be measured. The sensor has a transmitter and a receiver. The time taken by the ultrasonic wave to travel to the object and reflect back is measured and by that distance of the object is calculated using speed of sound wave in air medium. The sensor has 4 pins (VCC, GND, TRIG, ECHO). It operates on +5v and functions from 2cm to 400cm [4].

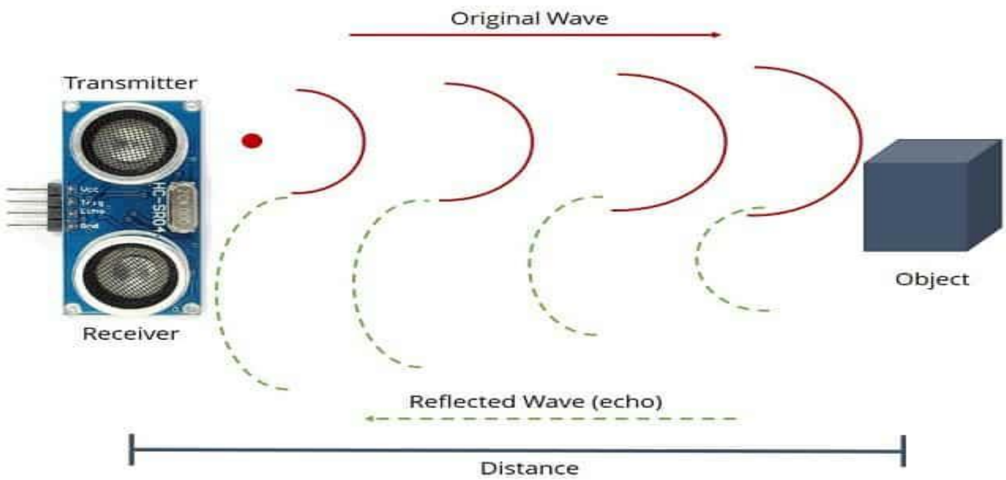

Figure 2: Working Principle of HC-SR04 Ultrasonic Sensor [8].

Proposed Work:
With self-driving cars, robots, automated drones and smart surveillance, the world depends much nowadays on detecting obstacles reliably and efficiently. Here, the proposed system utilizes YOLOv8, which is very efficient in real-time object spotting. This computer vision model is paired with HC-SR04 ultrasonic sensors for distance measurement of the obstacles. Under this scenario, this sensor fusion hybrid technology provides a strong and precise

solution that can find and measure impediments with great accuracy, even in a dynamic environment.

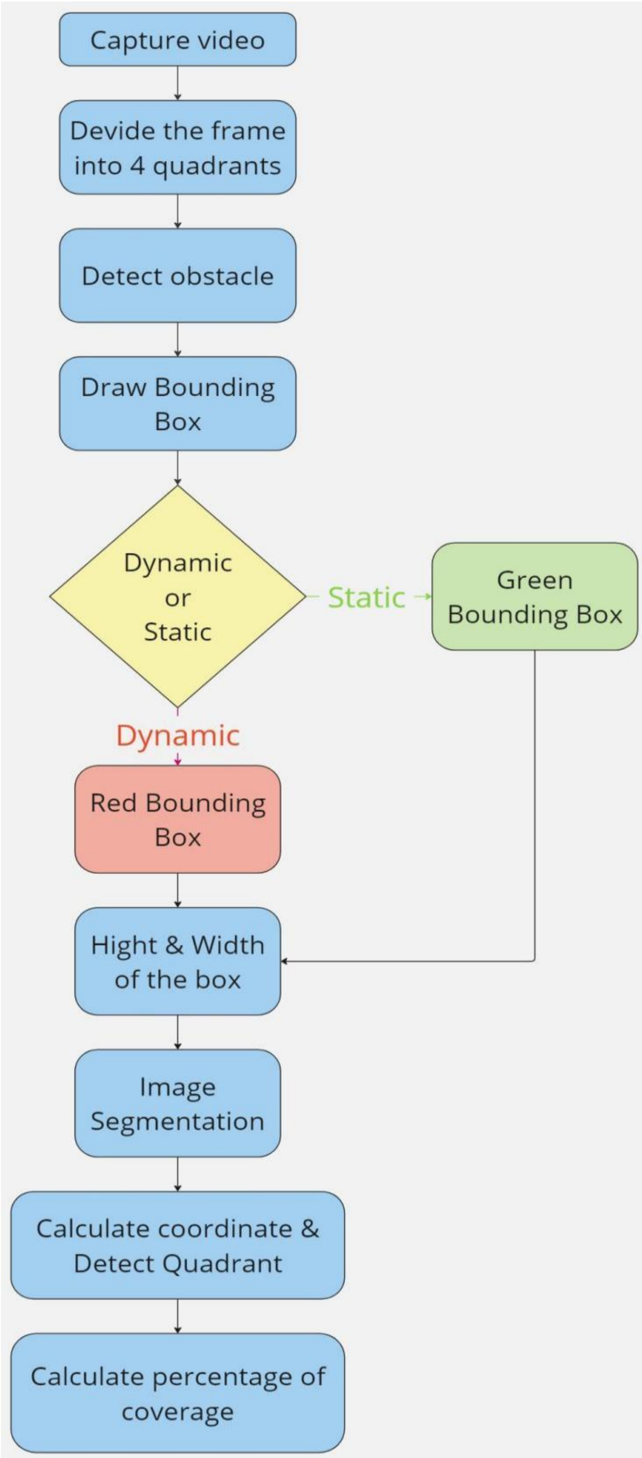

Figure 3: Workflow Diagram

Implementation: A framework for obstacle detection and tracking in UAVs has been implemented. The system for obstacle detection and tracking employs the YOLOv8 model with segmentation leveraging computer vision and machine learning methodologies to achieve real-time obstacle identification. Along with the segmentation coordinates, dimensions, percentage of coverage and status of movement is calculated. Additionally, it measures the range of obstacles using an ultrasonic sensor HC-SR04 based on the principles of SONAR from the drone.

First YOLOv8 is loaded to initialize the system. Video is captured through webcam using the OpenCV video capture function, and Tkinter is used to create the interface. The pretrained model is used to detect the obstacles and then the image segmentation is used to bound the obstacles precisely according to their shape within video streams. A Tkinter interface is used to display processed frames in real-time.

Every captured frame is processed to detect obstacles and analyze their properties. This work is done repeatedly in the loop. Every frame is then sent to the YOLOv8 model for obstacle detection Identification and segmentation. The architecture returns bounding boxes and masks on the detected obstacles. The bounding boxes provide coordinates, height, and width of each obstacle. Masks, on the other hand marks portray the exact shape and the portion occupied by the obstacle in the frame. Bounding boxes refer to rectangular enclosures surrounding each detected obstacle. The coordinates of the top-left corner and the bottom-right corner of the box are (x_1,y_1) and (x_2,y_2) accordingly. From this we can calculate the height and width Mathematically:

$$Height = y_2 - y_1$$
$$Width = x_2 - x_1$$

The image segmentation highlights the pixels occupied by each obstacle. The masks calculate the area of obstacles and indicate which quadrants are obstructed and their percentage of coverage. Mathematically:

$$Mask\ Area = \sum (\text{non-zero pixels in mask})$$

The frame is divided into four quadrants: the upper-left, upper-right, lower-left and lower-right sections. The obstructed areas under the mask are calculated for each of the quadrants.

$$Section\ Height = Frame\ Height/2$$
$$Section\ Width = Frame\ Width/2$$

The frame is divided into four quadrants. Each quadrant's non-zero pixels (obstructed area) are calculated and from that the percentage of obstruction is determined. Mathematically:

$$\text{Top-Left Obstruction} = \frac{\sum (non\ zero\ pixels\ in\ top\ left\ quadrant)}{section\ width \times section\ height} \times 100\%$$

$$\text{Top-Right Obstruction} = \frac{\sum (non\ zero\ pixels\ in\ top\ right\ quadrant)}{section\ width \times section\ height} \times 100\%$$

$$\text{Bottom-Left Obstruction} = \frac{\sum (non\ zero\ pixels\ in\ bottom\ left\ quadrant)}{section\ width \times section\ height} \times 100\%$$

$$\text{Bottom-Right Obstruction} = \frac{\sum (non\ zero\ pixels\ in\ bottom\ right\ quadrant)}{section\ width \times section\ height} \times 100\%$$

If the obstruction is greater than the threshold value, then the system generates a warning and if there is no hindrance in a quadrant then also the system will notify that the specific quadrant is free to move.

To determine if the obstacle is in motion or not the current position is compared with the previous frame. If the displacement is greater than a certain threshold level, then the obstacle is considered dynamic and bounded with red color else the obstacle is bounded with green bounding box. This distinction between static and dynamic obstacles is crucial for drone navigation.

$$Movement = \left((x_{current} - x_{previous})^2 + (y_{current} - y_{previous})^2 \right)^{1/2} > Movement\ Threshold$$

The height, width and centroid coordinate is calculated through the coordinates of the bounding box.

$$Height = y_2 - y_1$$

$$Width = x_2 - x_1$$

$$Centroid = \frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2}$$

The limitation with the computer vision model is that it cannot be trained for all the objects, and the processing is in 2-dimensional space so to calculate the distance of the obstacle from the drone an extra sensor is required. Here HC-SR04 ultrasonic sensor is used to achieve the goal. The Trigger pin emits sound wave at 40KHz frequency. The reflected wave form the obstacle is received by the Echo pin. Arduino Uno micro controller is assigned to measure the time lag between the emission and the receiving the wave and from that the distance of the obstacle is calculated using the speed of sound in air medium. On the measured time the sound wave travels in both directions, for that the distance is divided by 2 in order to get the one-way distance.

$$Distance = \frac{Time \times Speed\ of\ sound\ in\ air}{2}$$

Result and Discussion:

A comprehensive obstacle detection and tracking system for drones is successfully developed. The system achieves multiple outputs crucial for safe and efficient navigation. Coordinates of the obstacle are extracted to pinpoint the exact location within the drone’s field of view. Bounding box and masking over the detected obstacle highlight and outline each obstacle to know their size and shape. The system computes the height and breadth of the obstacle. The system determines the location of the obstacle quadrant wise that allows the drone or the remote pilot to plan a strategic path. Moreover, the system computes the percentage of the quadrant covered by the obstacle, thus showing the level of obstruction. If the percentage of obstruction is too high, then the system generates a stop message indicating that there is not enough space to travel. Then the system distinguishes between static and dynamic objects and highlights them with different colored bounding boxes. This system also uses ultrasonic sensors to detect the distance of the obstacle from the drone that increases safety and accuracy in the navigation process. Somehow the system manages to detect the semitransparent objects as well.

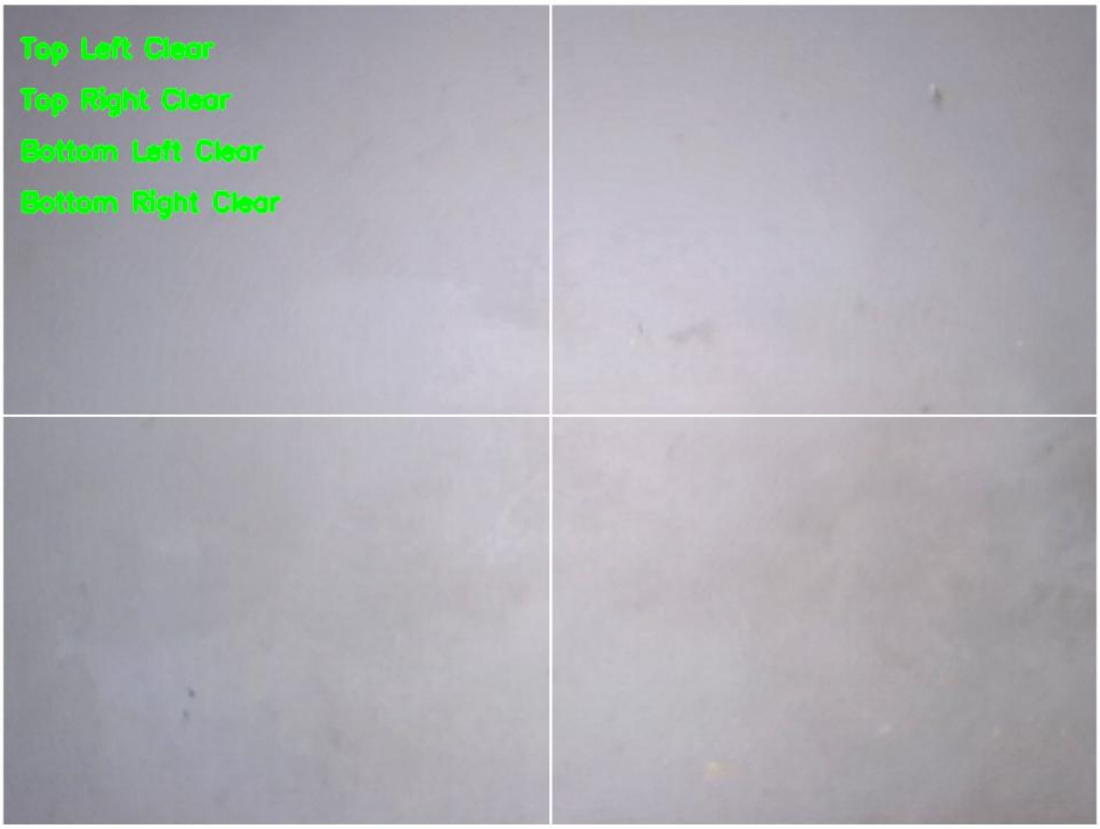

Fig-4: No obstacle situation

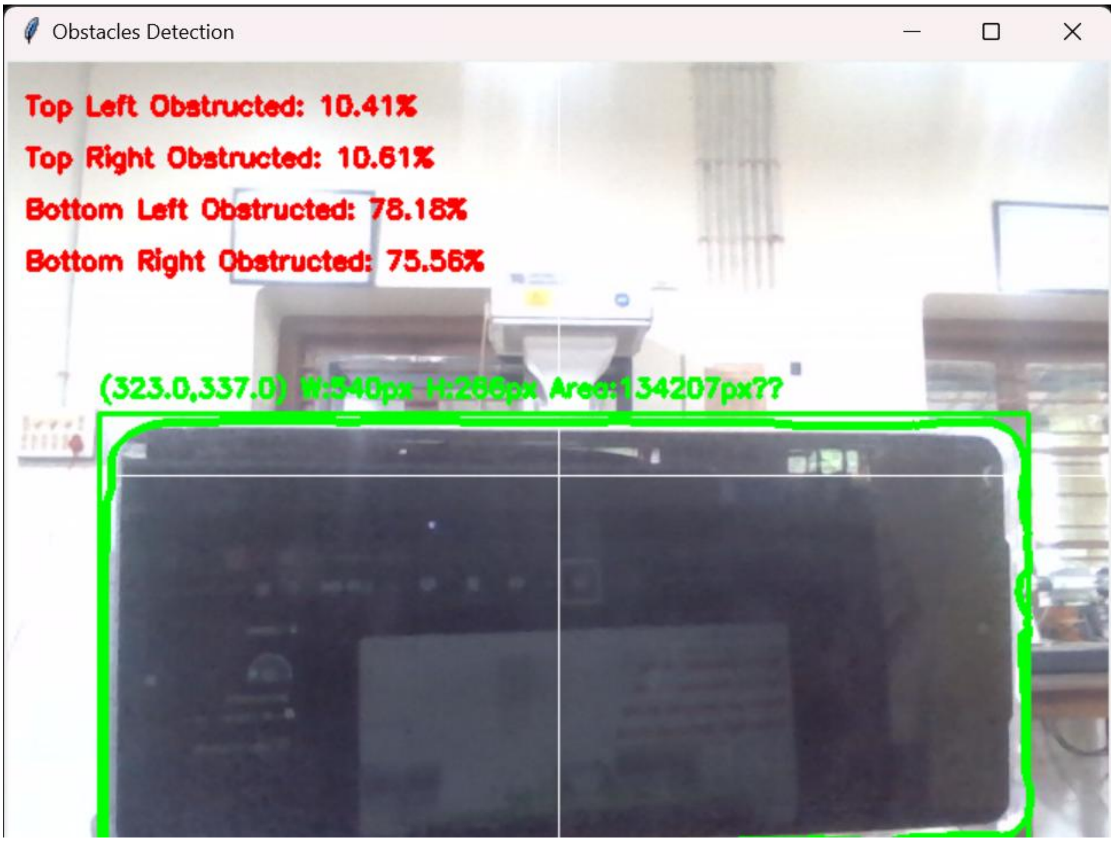

Fig-5: Symmetrical Obstacle

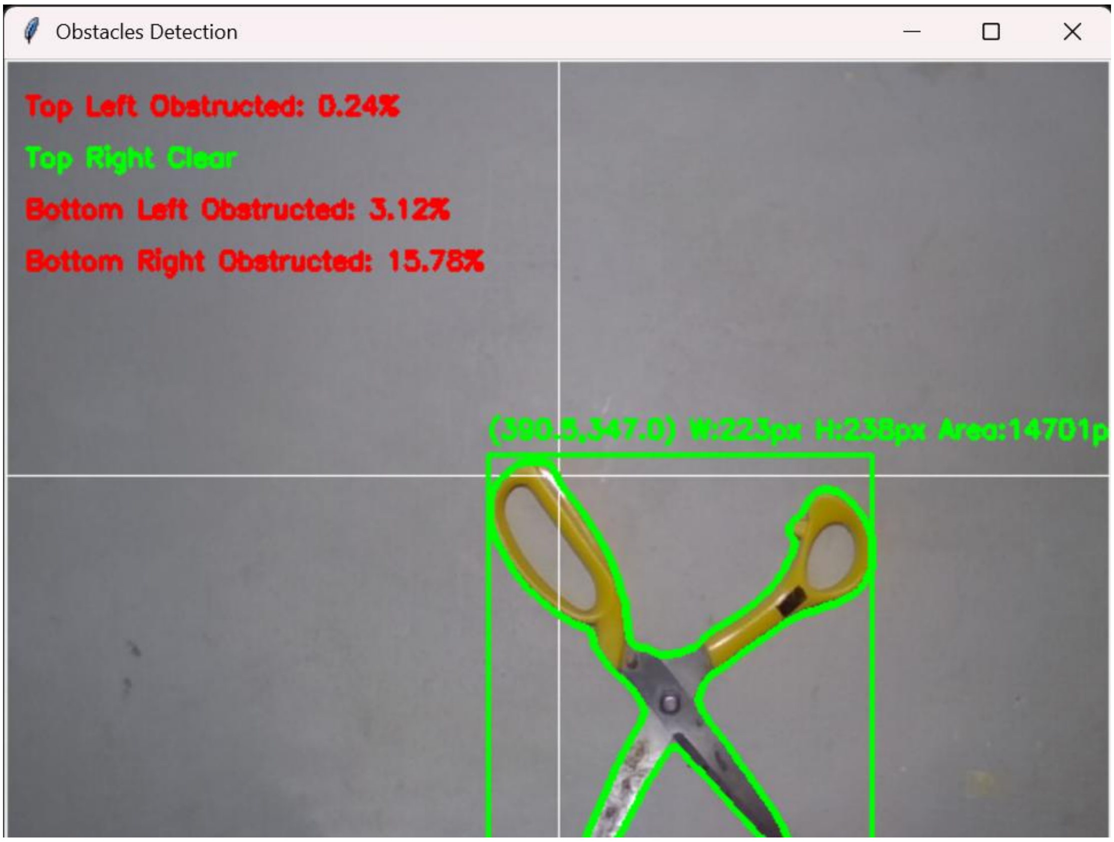

Fig-5: ‘X’ shaped obstacle

Fig-6: Uneven obstacle

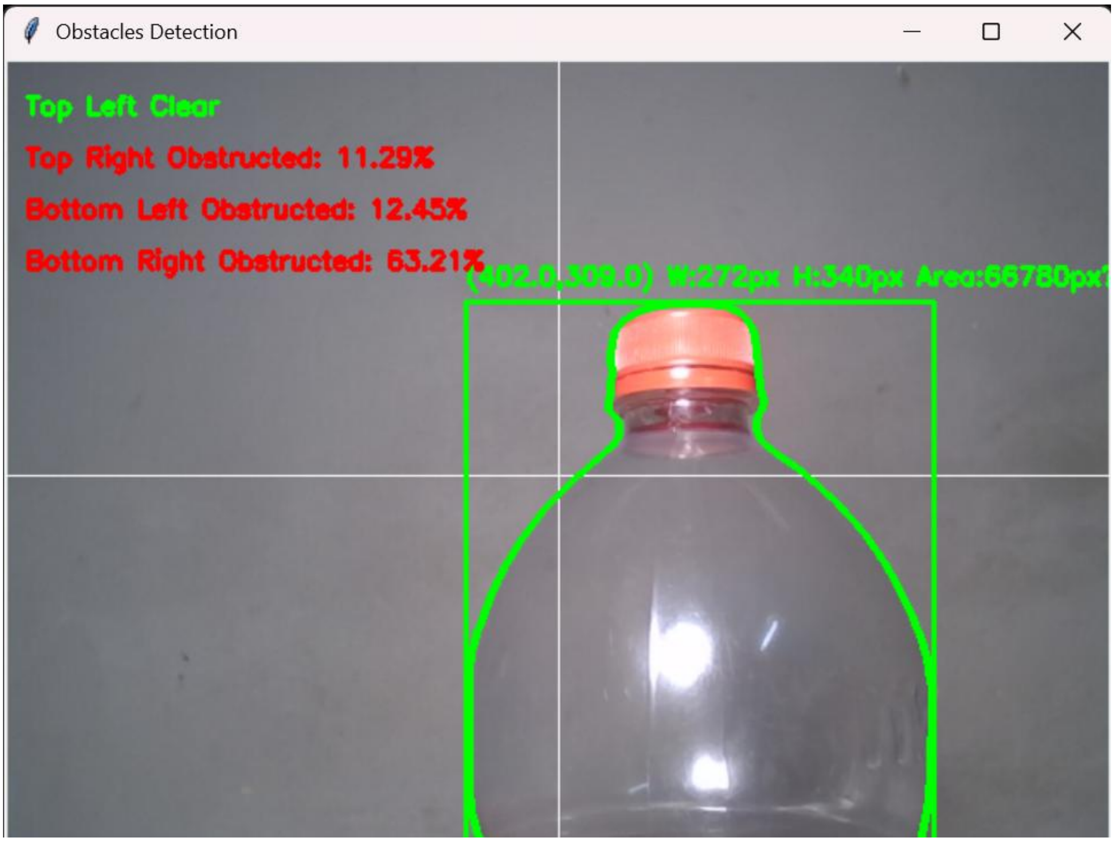

Fig-7: Semitransparent obstacle

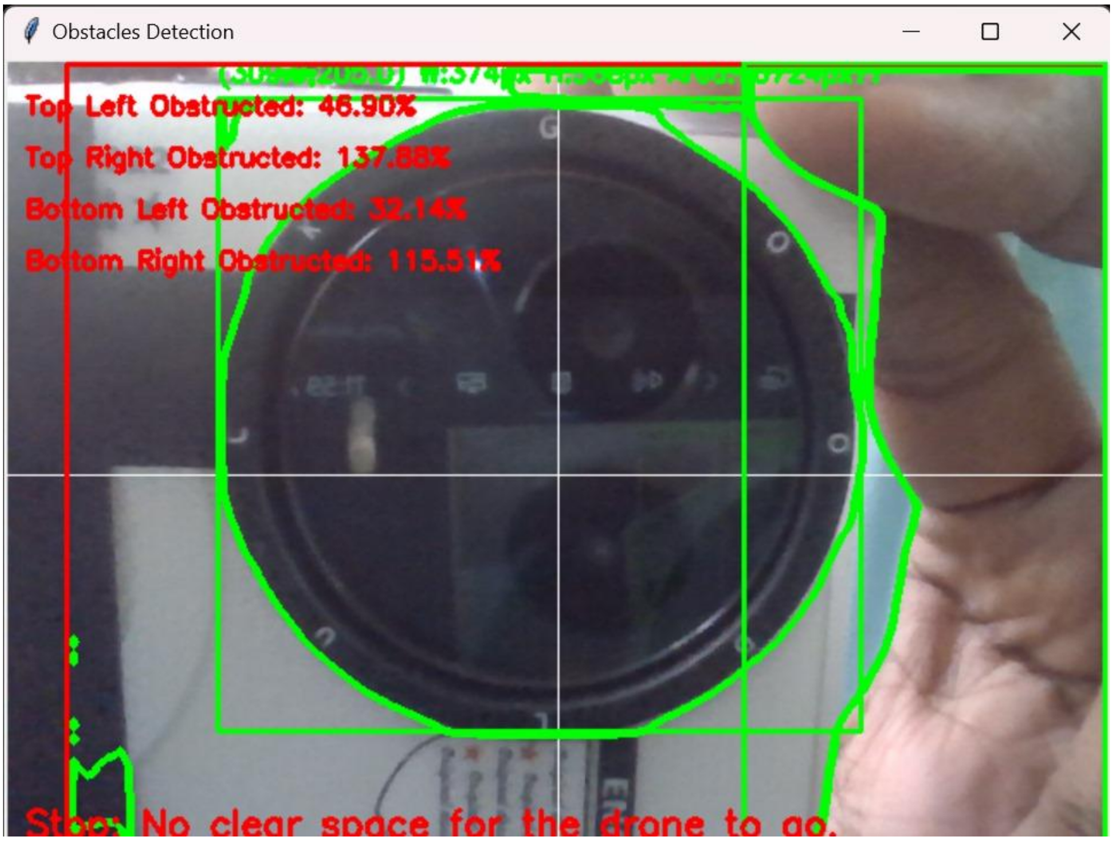

Fig-8: Large obstacle (No enough space)

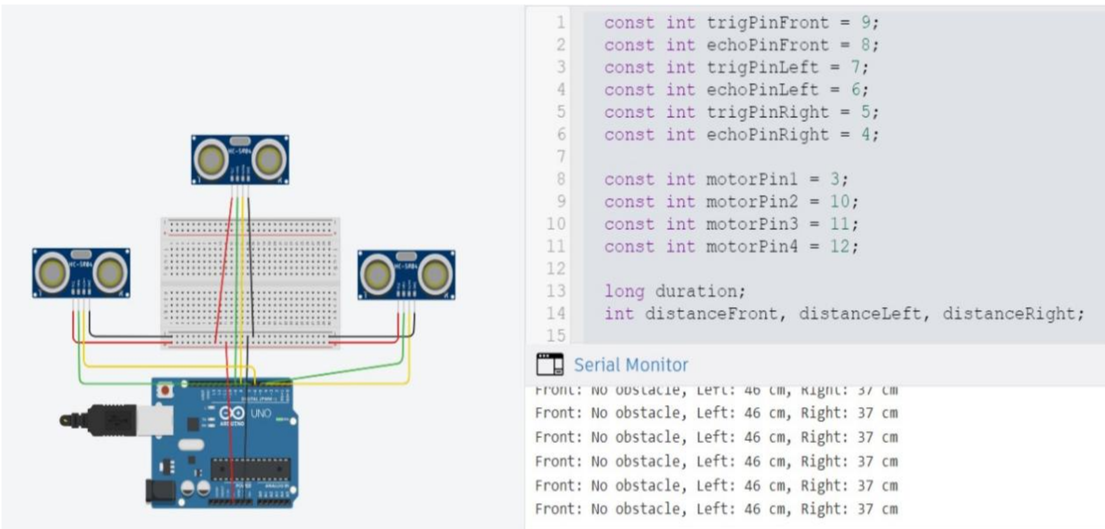

Fig-9: Implementation of Ultrasonic sensor

Table 2: Comparison of largely used object detection architectures in the context of recall, precision, F1-score.

MODEL	Recall	Precision	F1-Score
SSD	70–75%	72-76%	71-75%
Faster R-CNN	75-80%	80-85%	77-82%

Mask R-CNN	78-82%	82-86%	80-84%
RetinaNet	72–77%	78-82%	74-79%
Proposed work (Using YOLO v8)	80-85%	85-90%	82-87%

The proportion of correctly predicted instances (True Positives) out of all actual positive instances (True Positives + False Negatives) is calculated using Recall. It indicates how well the model detects all relevant objects. A higher recall means the model successfully detects more relevant objects with fewer false negatives.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

Precision measures how many of the detected objects are correct. High precision means fewer false positives.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positive}$$

F1-Score measures balance between precision and recall. High F1-score indicates the model has a good balance of detecting most objects (high recall) while also making few mistakes (high precision).

Table 3: Comparison of largely used object detection architectures in the context of accuracy.

MODEL	mAP (COCO, IoU=0.50)	mAP (COCO,IoU=0.50:0.95)
SSD	77–80%	37–40%
Faster R-CNN	82%	42–45%
Mask R-CNN	83%	43–47%
RetinaNet	80–85%	39–42%
Proposed work (Using YOLO v8)	85–90%	50–55%

Here: **mAP IoU=0.50**: Measures the average precision when Intersection over Union (IoU) threshold is 0.50.
mAP IoU=0.50:0.95: A stricter evaluation metric averaging over IoU thresholds from 0.50 to 0.95.
The Mean Average Precision values are compared in terms of object detection capability on general benchmark like COCO dataset.

mAP is derived from precision-recall curve.

$$IoU = \frac{Area\ of\ overlap}{Area\ of\ union}$$

IoU = 1 indicates perfect match and IoU = 0 indicates no match at all

Table 4: Comparison of largely used object detection architectures.

Feature	SSD	Mask R-CNN	Faster R-CNN	RetinaNet	Proposed system Using (YOLOv8)
FPS	Very fast 20-40 FPS	Low 3-5 FPS	Moderate 5-7 FPS	Fast 15-20 FPS	Very fast 50+ FPS
Inference Time	10-15 ms	80-120 ms	90-150 ms	60-90 ms	5-10 ms
Model size	~100 MB	~250 MB	~220 MB	~150 MB	~8 MB
Training Time	~1-3 hrs	~10+ hrs	~6 hrs	~5 hrs	~1-2 hrs

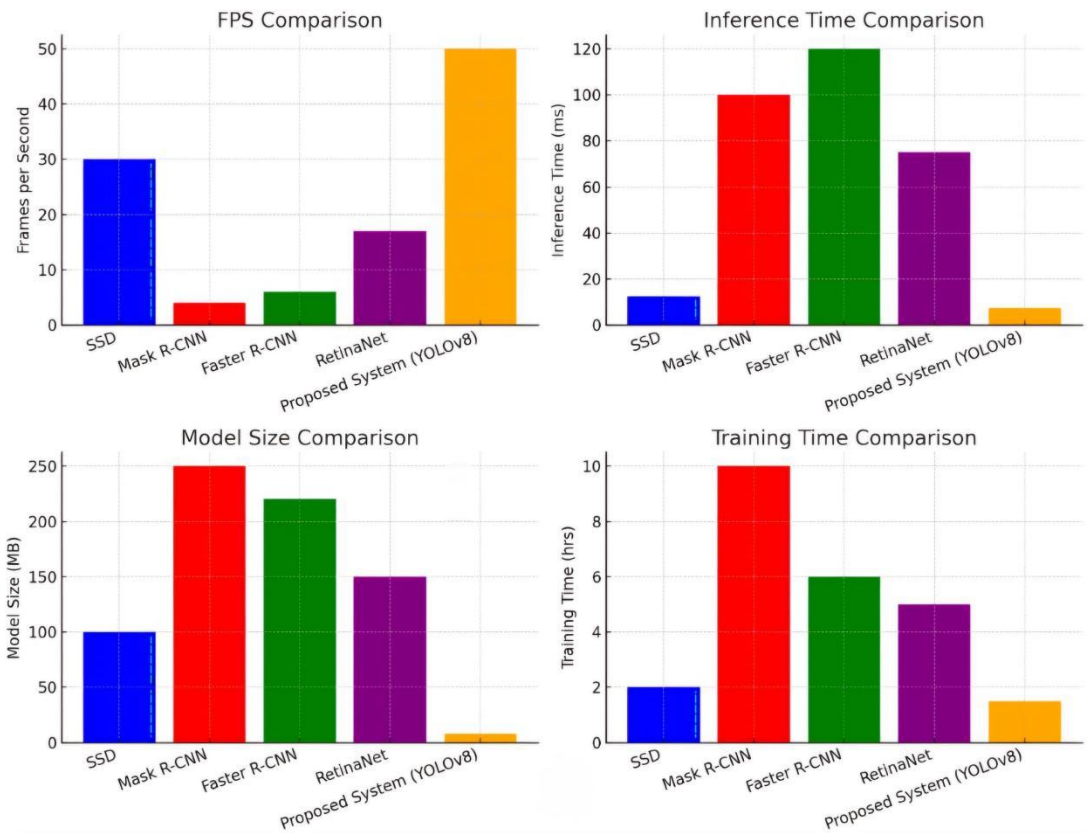

Fig-10: Comparison of different object detection models

Inference time is the time takes to proxess a single image. Low inference time is very crucial for real time applications. By this comparison we can observe that obstacle detection for UAV and drones, YOLOv8 would be best fit in terms of real time performance and accuracy.

The key highlight is the model detects all types of obstacles with high recall, precision and speed. It accurately points outs the obstacle and precisely segments the 'X' shaped and uneven obstacles clearly highlighting their actual shape and size. It locates the obstacle exactly following the segmented region not the bounding box. The information like the coordinate, position, height, width and the area coverage also work decently. The stop message also pops up if there is no adequate space for the drone to go forth. The 3 Ultrasonic sensors integrated with Arduino UNO are also able to measure the distance of the obstacle.

Conclusion:

In this paper obstacle detection using YOLOv8 is successfully implemented for enhancing the accuracy and preciseness of obstacle detection and tracking with image segmentation. The concept of image segmentation is inspired by mask R-CNN. In this model for obstacle identification a COCO databased pre-trained model is used. The bounding box with the mask efficiently gives detailed information about the shape and size of the obstacle. Along with that height, the width of the obstacle is calculated w.r.t the dimension of the bounding box. Coordinates of the centroid of the obstacle is found by taking the intersection of the diagonals of the bounding box which helps to determine the location of the obstacle. Then the system judges the object as dynamic or static, which is very crucial for autonomous navigation. The frame is divided into quadrant so that the obstacle can be located more comprehensively. The obstructed area is calculated by taking summation of non-zero pixels in the view field. The area calculation and percentage of coverage information gives insights about if the drone can go through the obstacles or not. This computer vision model is infused with ultrasonic based obstacle finding and ranging. 3 HC-SR04 ultrasonic sensors are diploid in the 3 sides of the drone to determine the distance of the obstacle from the drone. If there is an obstacle within the certain range then the sensor will send information to the Arduino Uno micro controller and using the value of time taken to travel the sound wave in both directions the micro controller calculates the distance of the closest obstacle, which provides a safe range for the drone. Future work can be training the model with a larger dataset with more varieties of objects and more numbers of images of each object class, making the model more robust and applicable for unfamiliar environment.

Reference:

1. Ahmed Ben Atitallah et at, "Embedded implementation of an obstacle detection system for blind and visually impaired persons' assistance navigation." https://www.sciencedirect.com/science/article/abs/pii/S0045790623001386

2. "Automated Detection of Atypical Aviation Obstacles from UAV Images Using a YOLO Algorithm" by Marta Lalak and Damian Wierzbicki.

3. "Deep Learning-based Monocular Obstacle Avoidance for Unmanned Aerial Vehicle Navigation in Tree Plantations" by H. Y. Lee et al, Journal of Intelligent & Robotic Systems (2021)

4. R.Vairavan, S.Ajith Kumar et al, "OBSTACLE AVOIDANCE ROBOTIC VEHICLE USING ULTRASONIC SENSOR, ARDUINO CONTROLLER", International Research Journal of Engineering and Technology (IRJET), vol: 5 2018

5. Sanjukumar NT and P Rajalakshmi, "Obstacle Detection and Collision Avoidance on UAV using Rangefinder Sensor with Kalman Filter Technique", 2022 IEEE Global Conference.

6. "DS-YOLOv8-Based Object Detection Method for Remote Sensing Images" by Lingyun Shen; Baihe Lang; Zhengxun Song, https://ieeexplore.ieee.org/document/10310143

7. Redmon, J., Divvala, S., Girshick, R., and Farhadi, A., "You Only Look Once: Unified, Real-Time Object Detection," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.
8. Arduino Ultrasonic Sensor Overview and Tutorials By Shawn .
https://www.seeedstudio.com/blog/2019/11/04/hc-sr04-features-arduino-raspberrypi-guide/
9. S. Suherman, R. A. Putra and M. Pinem, "Ultrasonic Sensor Assessment for Obstacle Avoidance in Quadcopter-based Drone System," 2020 3rd International Conference on Mechanical, Electronics, Computer, and Industrial Technology (MECnIT), Medan, Indonesia, 2020.
10. “Distance Measurement Using Ultrasonic sensor & Arduino” Pavan Pawar , Avinash Giri , Jagdish Taur , Rutika Hajare , Atharava Apurva, 2023 IJCRT | Volume 11, Issue 12 December 2023 | ISSN: 2320-2882
11. ijece.iaescore.com
12. Aswini N, Uma S V. "Custom Based Obstacle Detection Using Yolo v3 for Low Flying Drones", 2021 International Conference on Circuits, Controls and Communications (CCUBE), 2021
13. Lingyun Shen, Baihe Lang, Zhengxun Song. "DS-YOLOv8-Based Object Detection Method for Remote Sensing Images", IEEE Access, 2023
14. Mingzhou Liu, Xin Xu, Jing Hu, Qiannan Jiang. "Real-time unstructured road detection based on CNN and Gibbs energy function", Journal of Intelligent & Fuzzy Systems, 2022