Materials Today: Proceedings 72 (2023) 1817–1824

Contents lists available at ScienceDirect

Materials Today: Proceedings

journal homepage: www.elsevier.com/locate/matpr

Object following robot based on AI/ML

Anand Kamble, Anirban C. Mitra, Aniket Tathe *, Suyash Kumbharkar, Atharva Bhandare

Department of Mechanical Engineering, Modern Education Society's College of Engineering, Pune 411 001, India

ARTICLE INFO

Article history:
Available online 11 October 2022

Keywords:
Object following
Deep neural network
OpenCV
Raspberry Pi
WebSocket

ABSTRACT

Object following is an emerging field in the Computer Vision. It's not like object detection. Object detection only covers the identification part while object following also involves tracking the path of the identified object. Object following can be used to follow any object or human. Following a selected object in an environment of multiple moving objects can be implemented in various fields and has multiple possible applications. Such robot can be useful in hotels for delivering the food at the required destination and assisting the workforce. This type of robot can also be used in warehousing application where the robot can be loaded at entrance and unloaded at the destination. In this paper, we introduce an object following robot in an environment consisting of multiple non-stationary objects while giving the user manual control and live feed of the robot through a web user interface.

Selection and peer-review under responsibility of the scientific committee of the 2nd International Conference and Exposition on Advances in Mechanical Engineering.

1. Introduction

In current times much research is going on in the field of object tracking and object recognition. Object following algorithms can be used in many fields, like video surveillance and supervision on a vicinity. In the future, object following service robots will be able to provide a lot of services and assistance. The core part is object tracking with following the object physically in real time. At present there are many robots which are used to assist humans in various tasks which makes the job simpler and efficient. Many human following robots which are seen online mostly make use of ultrasonic sensor but as the ultrasonic sensor just returns distance and we cannot rely on it for real world solid application as it doesn't distinguish between different objects. This results in a high false alarm ratio. Hence a reliable system should be introduced which can distinguish between human and other objects and also has a low false alarm ratio. For this machine learning comes handy in which deep learning provides various objection detection algorithms. A web user interface can also be used to monitor the robot. There are three fields applied in this project one is machine learning for object detection and robotics for motion and web development in user interface. There are various object detection algorithms some of which are Region-based Convolutional Neural Networks (RCNN) [1], Faster RCNN [2], Single Shot Detector (SSD) [3] and You Only Look Once (YOLO) [4]. Among them, in terms of accuracy Faster-RCNN and SSD perform better as compared to rest of the algorithms, when one needs speed over accuracy YOLO gives better result. SSD and Mobile Nets combined prove to be more efficient in object detection and following. The algorithm used in the robot is SSD MobileNetv3 large which is trained on COCO dataset [6]. To allow real-time, full-duplex communication WebSocket protocol is used. This is possible with the help of SocketIO library. After comparing Arduino UNO, ESP 32[8] and Raspberry Pi, Raspberry Pi was used as Raspberry Pi provides much better computation results required for image processing as compared to Arduino UNO and Wi-fi module in one single device. For controlling the motors, dual channel motor driver is used and since total 4 motors are being used these motors are connected in parallel in pair to the motor driver. Ultrasonic sensor is used in order to avoid collision of the robot with the objects present in the vicinity. Since the battery provides an output voltage of 12 V and 9A and Raspberry Pi 3 Model B only requires 5 V to operate LM2596 DC-DC Buck Converter is used to adjust the voltage. The camera is connected with Raspberry Pi 3B. Total four Johnson HS850 DC Motor with 100 RPM and 1 N/m torque output are used as motors.

2. Materials and methods

2.1. Object detection algorithm

Machine learning is a subset of Artificial Intelligence and deep learning is a subset of machine learning. In deep learning there are various object detection algorithms which function on different

* Corresponding author.
E-mail address: aniket.motog3.tathe@gmail.com (A. Tathe).

https://doi.org/10.1016/j.matpr.2022.09.577
2214-7853/
Selection and peer-review under responsibility of the scientific committee of the 2nd International Conference and Exposition on Advances in Mechanical Engineering.

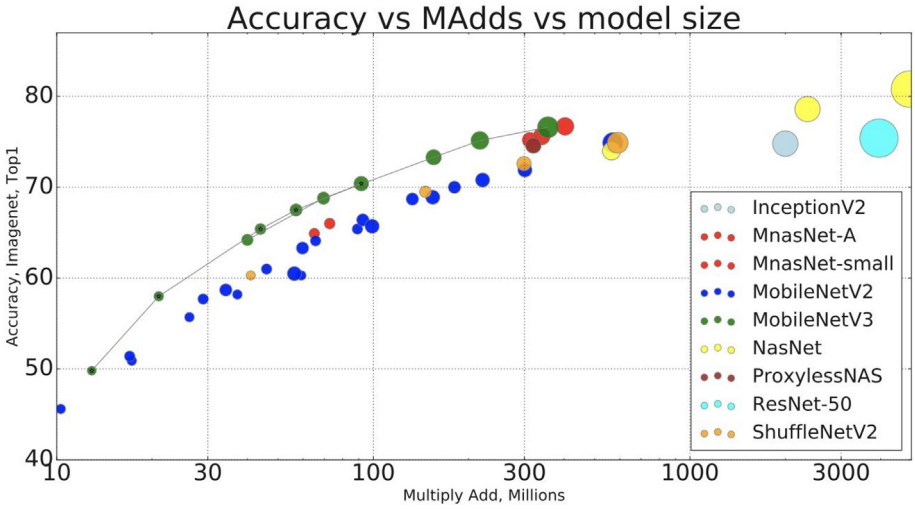

Fig. 1. Accuracy vs MAdds vs model size [9].

Table 1
Components.

Sr No	Components	Quantity
1	RASPBERRY PI 3 MODEL B	1
2	Battery	1
3	Ultrasonic sensor	1
4	LM2596 DC-DC Buck Converter	1
5	Camera	1
6	Johnson HS850 DC Motor	4
7	Motor Driver	1

logic. The selection criteria vary as per the application and needs of the user. Accuracy and speed are the two important factors to be considered while selecting and implementing a particular algorithm. The algorithm which gives the best accuracy in satisfactory time frame and memory needed as per the application is to be selected. MobileNet is a popular object detection algorithm which is well known for its compact size and high accuracy. Various versions are available in MobileNet. Among them compared to MobileNetV2, MobileNetV3-Large is more accurate on ImageNet classification by 3.2% also latency of MobileNetV3-large 15% less as compared to MobileNetV2. When MobileNetV3-Small and MobileNetV2 are compared we get similar results with MobileNetV3-Small being 4.6% more accurate and 5% less latency as compa red to MobileNetV2. On COCO dataset, both MobileNetV3-Large and MobileNetV2 perform approximately similar in terms of accuracy but in terms of speed MobileNetV3-Large performs 25% faster as compared to MobileNetV2 [9] (Figs. 1 and 2).

2.2. WebSocket

Communication is the exchange of information or data between two or more systems which can be roughly divided in two categories, synchronous communication and asynchronous communication. Asynchronous communication is implemented in HTML forms and also in back-end systems like database. Synchronous communication systems also called real time communication systems, consists of multiple passive recipients [10]. Ajax allows the web clients of HTTP communication system to asynchronously poll

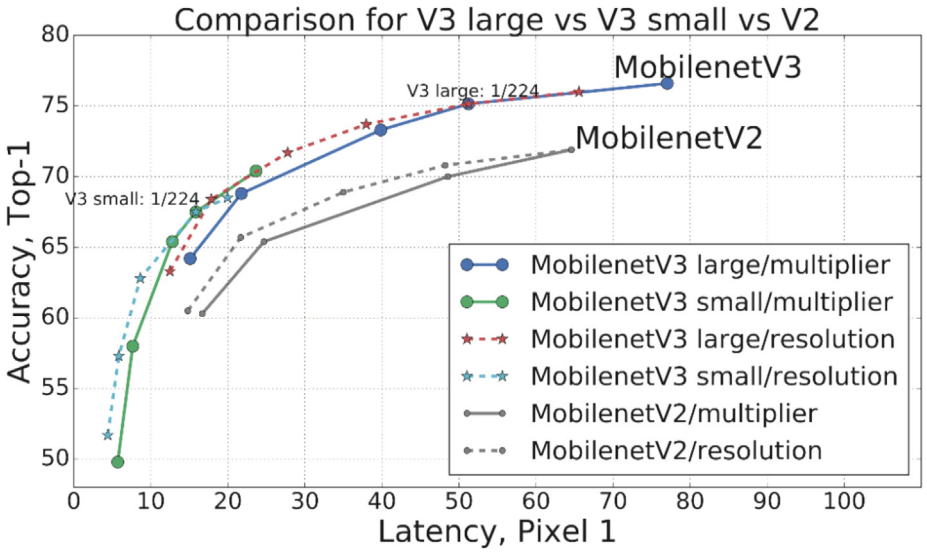

Fig. 2. Comparison for V3 large vs V3 small vs V2 [9].

the server side events while Comet allows "push"-style communication from server to client. WebSocket which is a HTTP communication system, provides bi-directional connection also called as full-duplex connection. This makes pulling and pushing the information between server and client possible. Hence WebSocket is more efficient while avoiding the connection issues of Comet or Ajax polling.

2.3. Raspberry-Pi 3 Model B

Raspberry-Pi 3B has a Quad core 64bit processor with 1 GB ram which is optimal because of its portability and power efficiency As the Raspberry Pi 3B has inbuilt Wi-fi module which is used to transfer data wirelessly. It has multiple USB type-A ports which is used to connect the camera so that it can capture images. Raspberry Pi also provides GPIO pins which are useful to connect other electronics devices such as sensors and motor drivers. The Raspberry Pi is a small single-board computer (SBC) which has integrated ARM central processing unit (CPU) and graphic processing unit (GPU). Raspberry Pi is being used in the robot to control the motors through GPIO pins and integrated Wi-fi module is used to achieve communication to the server (Table 1 and Figs. 2–6).

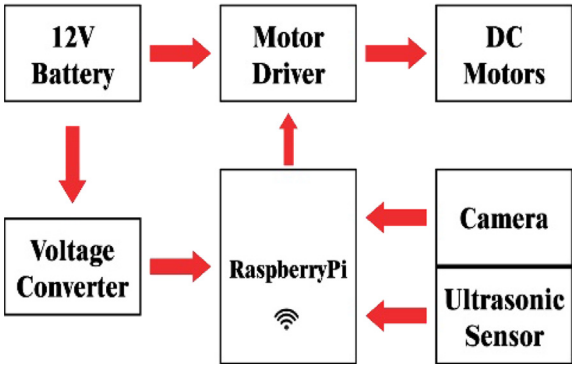

Fig. 3. Flowchart.

2.4. Working

The whole system can be divided in three parts:

(1) Robot
(2) Server
(3) Client

2.5 (. a) Robot

Raspberry Pi is mounted on the robot. The code is executed on Raspberry Pi. Robot side code handles all the communication between server and the robot and executes commands received from server for example command for movement. This code also captures video from the camera and sends it to the server in an encoded format which encodes image data to string using base64 encoding scheme for transmission. As the image processing and object detection adds huge amounts of delay if executed on Raspberry Pi the image processing task is performed on the server side. Raspberry Pi also executes the ultrasonic sensor's code which prevents robot from collision. It also has the code required for movement of the robot's movement manually whenever needed by the client.

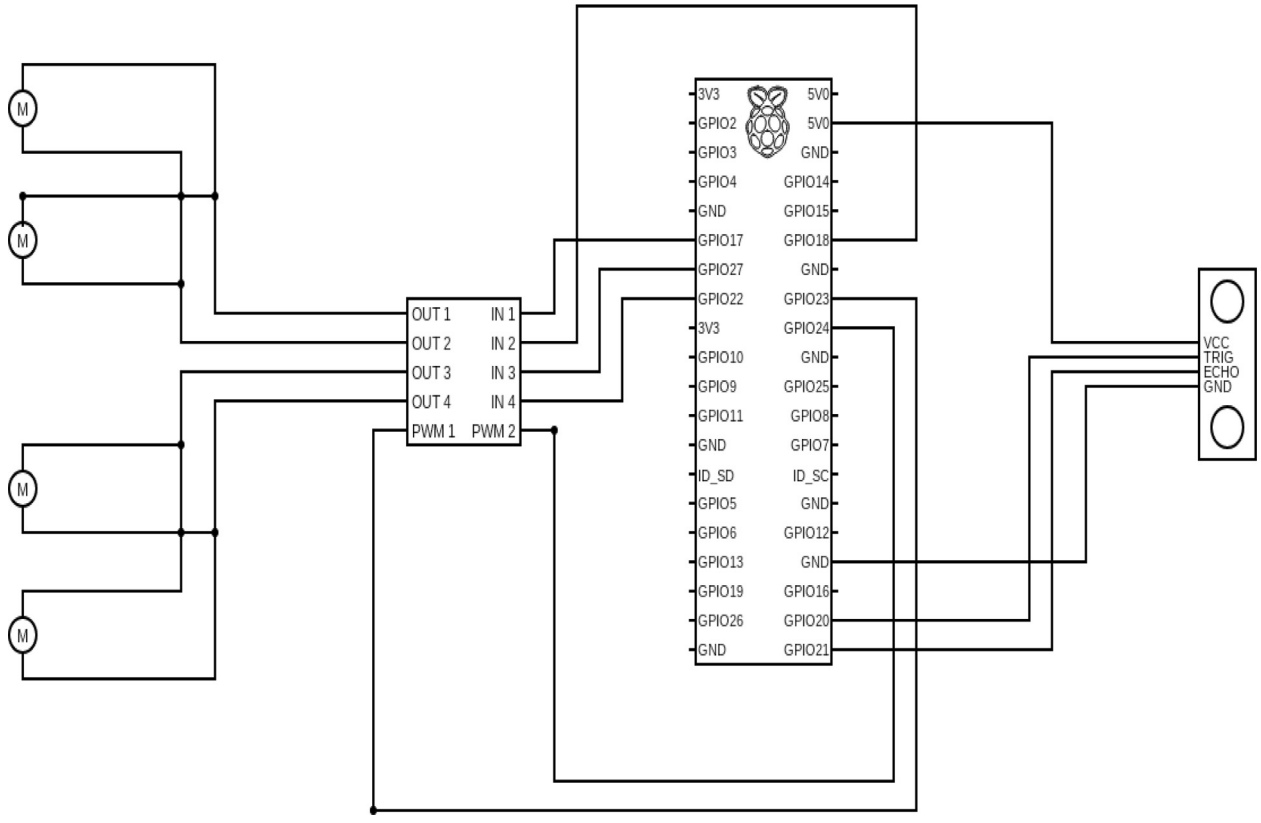

Fig. 4. Circuit Diagram.

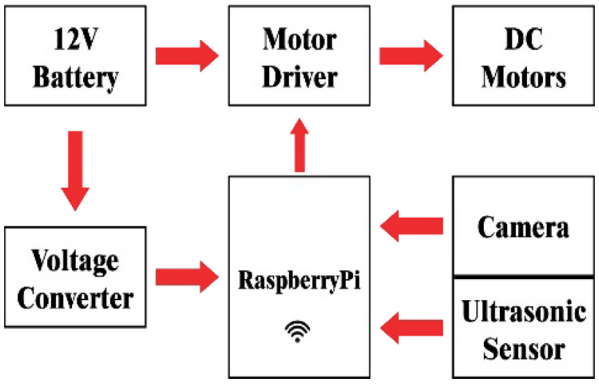

Fig. 5. 3d model.

2.6 (. b) Server

Server side's code is the most important as it handles image processing and object recognition. The images received from the robot are processed using library named OpenCV. This processing is done on the server to minimize the processing time required to process one frame of the video and allows the robot to react to the surroundings in real time. Server also handles the client side requests which includes commands for manual control of the robot and forwards it to the robot. It also serves the live feed from the robot to client which allows monitoring of the robot. Server uses a recursive algorithm which only accepts images when processing of previous image is done which synchronizes communication between the robot and server.

2.7(. c) Client

Made the frontend program using HTML, JavaScript and CSS. JavaScript is used to allow cross platform compatibility. This allows us to control the robot's movements and watch the live feed from the robot on any devices like Computers and Mobiles through a web browser. This allows the client to change the speed of the movements of robot and move it manually in forward, reverse, left and right directions. The code can handle multiple clients at the same time (Fig. 7).

3. Results and discussion

The robot has been made to follow a bottle for testing the algorithm and accuracy. Fig. 9 shows the robot has detected a file. Hence, it doesn't follow it and remains stationary but when the file is removed from the sight of robot, the bottle becomes visible to the robot. In Fig. 10, bottle can be seen on the user interface and when the machine learning algorithm performs object detection, the bottle gets detected which triggers the algorithm for motors and the robots starts moving in the direction of the bottle until it reaches within 20 cm and stops. The ultrasonic sensor is set in such a way that once the distance between robot and object is less than 20 cm the algorithm for motors is stopped to avoid collision. Fig. 11 since the distance between robot and bottle is 20 cm it stops. In Fig. 12, the bottle is shifted towards right side. Fig. 13 shows when the bottle is shifted to the right side the robot also turns in that direction and then starts following it until the distance between the robot and the bottle is 20 cm (see Fig. 8).

Based on above results we can also similarly use this robot to follow human beings. In case of multiple human beings in sight

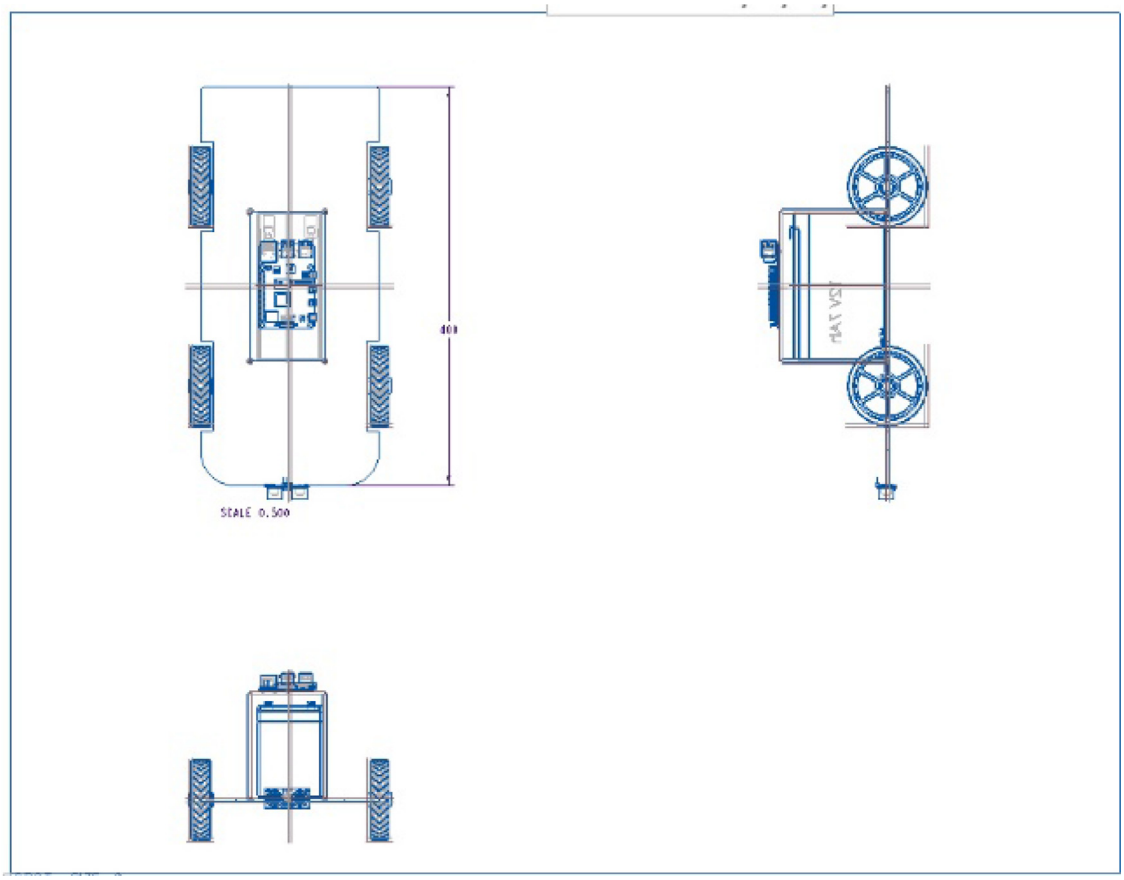

Fig. 6. Front, Top and side view of the robot.

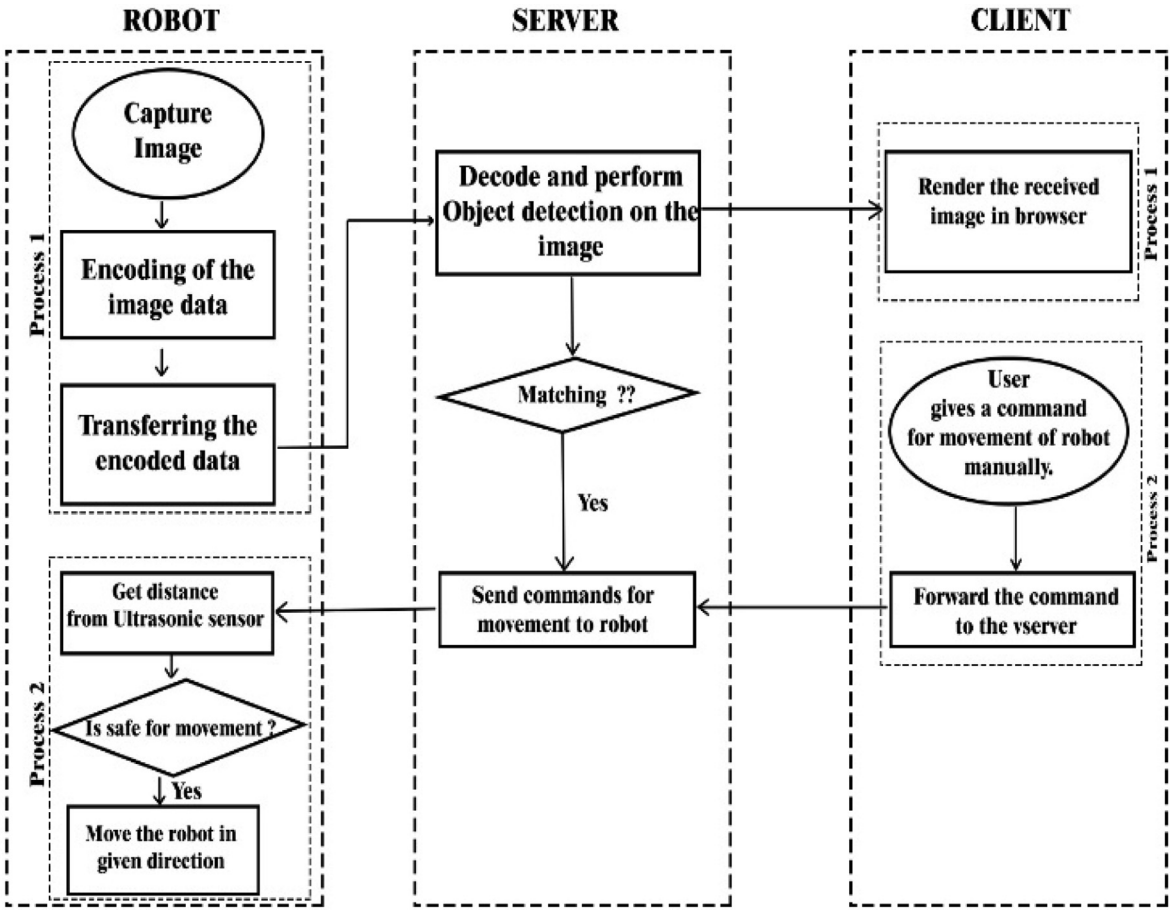

Fig. 7. Working.

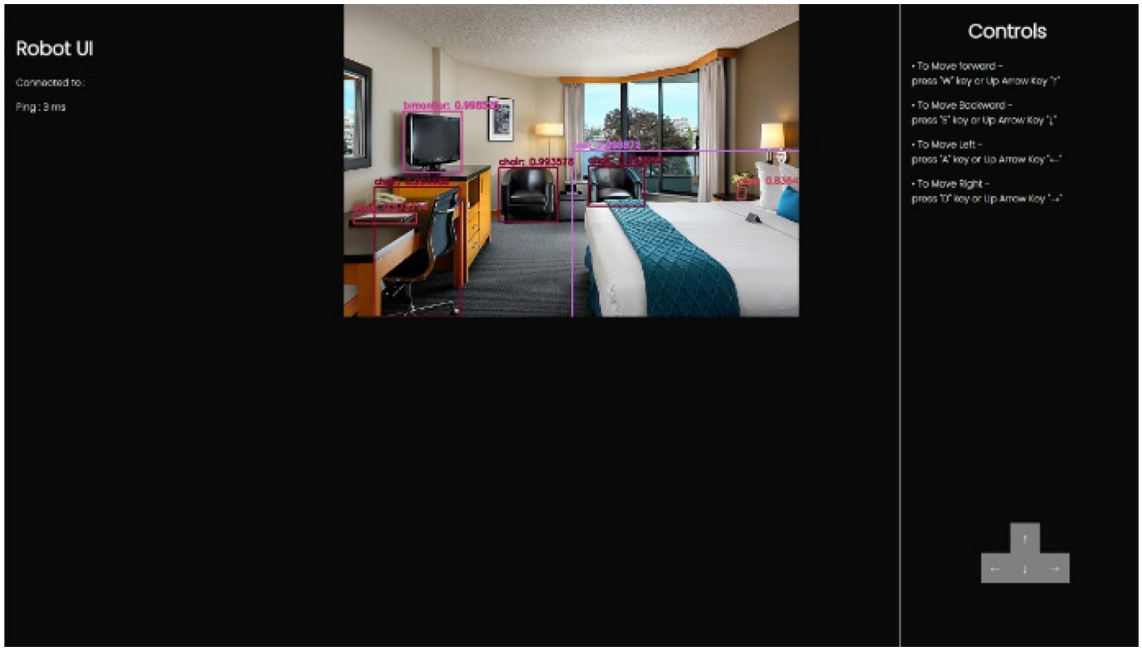

Fig. 8. User Interface.

the robot algorithm is designed in such a way that it will only follow the human being which has the greatest bounding box width i. e the closest human being in front of the robot in the feed.

The robots which are being used currently in restaurants in countries like Japan consists of line tracking robots which can only move in a predefined path. The robot mentioned above does not

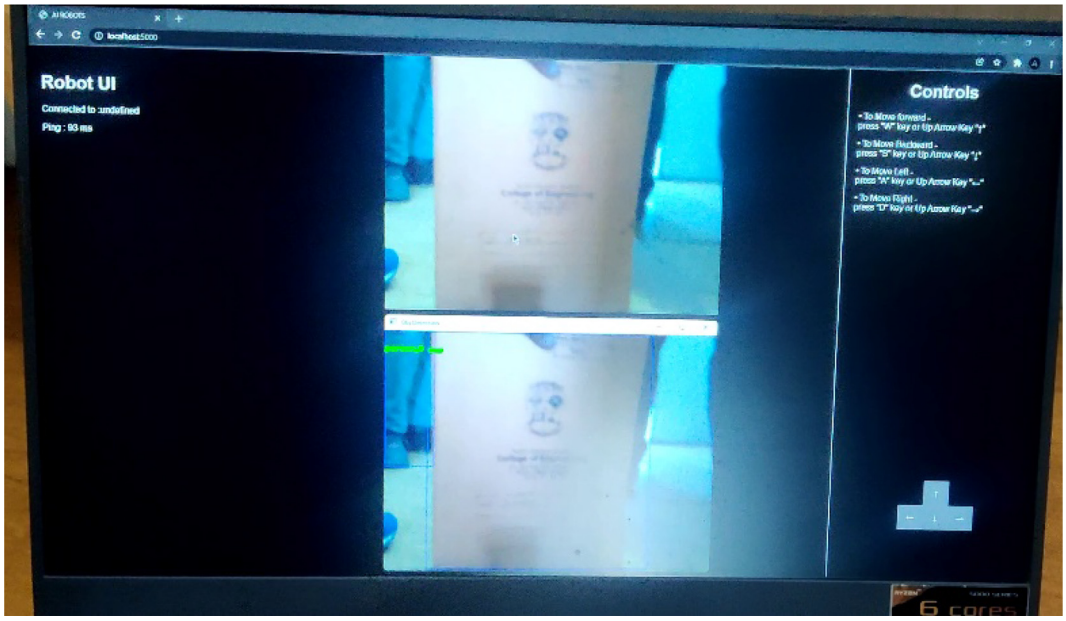

Fig. 9. Testing Object detection (a).

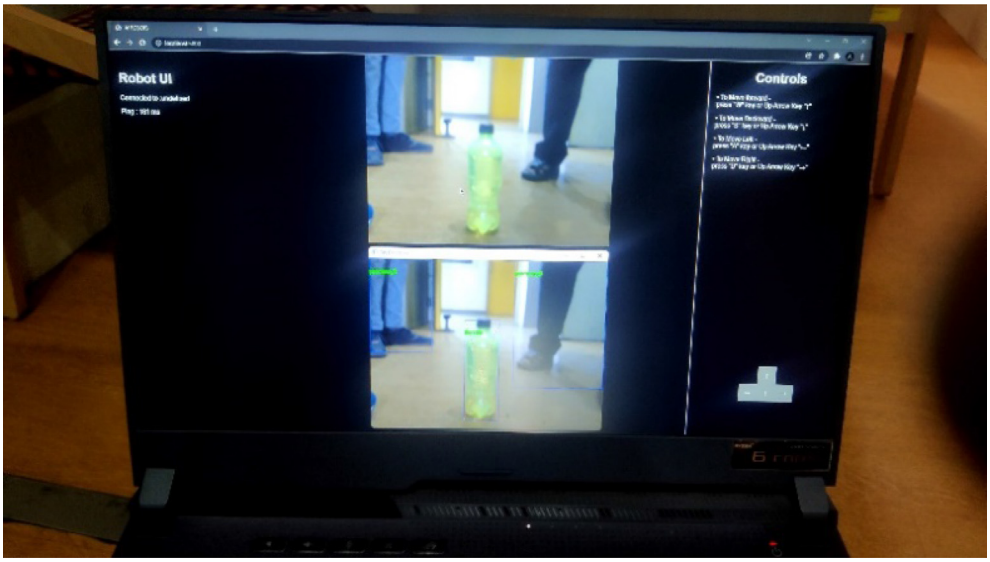

Fig. 10. Testing Object detection (b).

have such limit. It can move in any direction. It can follow the workers and assist them in serving the food if an empty compartment on the top is introduced in the assembly on which objects likes dishes, food, etc. can be kept. When multiple orders need to served, the food can be placed in the compartment of the robot and the robot can follow the waiter and once the necessary table is reached the food can be served easily by the waiter in one-go instead of multiple turns without the robot. This will reduce the workload on the workers which in turn improves overall efficiency and also customer satisfaction.

During warehousing when new stock is to be loaded into the warehouse the required material is carried by the human alone or with the help of a trolly or forklift. This robot can be used similarly in such situation with just a added compartment like trolly, etc on top of current assembly. Hence when the new stock arrives the robot can be loaded at entrance and can follow the human which needs to do the unloading and arranging the stock at the required destination. Hence the human just needs to reach the destination without any hassle and just do unloading. Such types of robots help when the required unloading destination is quite small or narrow and where large vehicle like forklift cannot enter and the only option there is manual loading and unloading by human. Since the robot is not huge it can reach destination easily and the human cand perform the unloading task efficiently.

Object detection is being performed on the object in the environment and the result can be seen on the user interface in real-time. Once detected, the robot starts following the specified object. This can be used in various areas like luggage carrying on airport, homeland security in which instead of deploying multiple CCTV for respective rooms, a single robot can be used for the same task. Similarly non-stationary object which is in motion can be kept under constant surveillance in an environment which will keep a

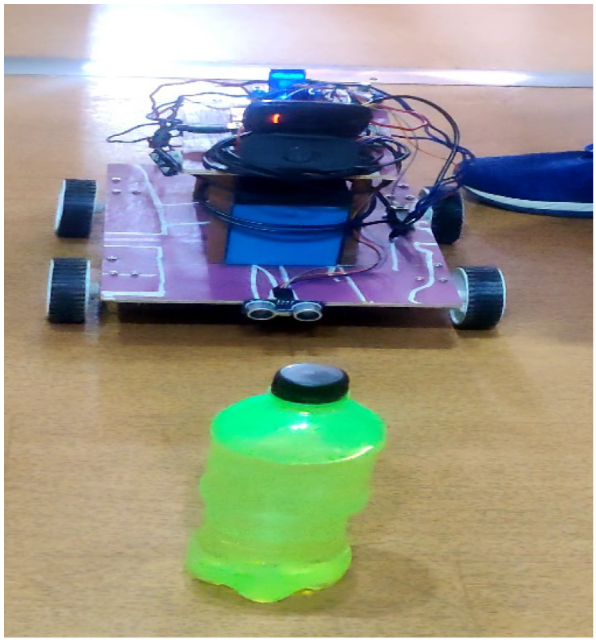

Fig. 11. Testing Object detection (a).

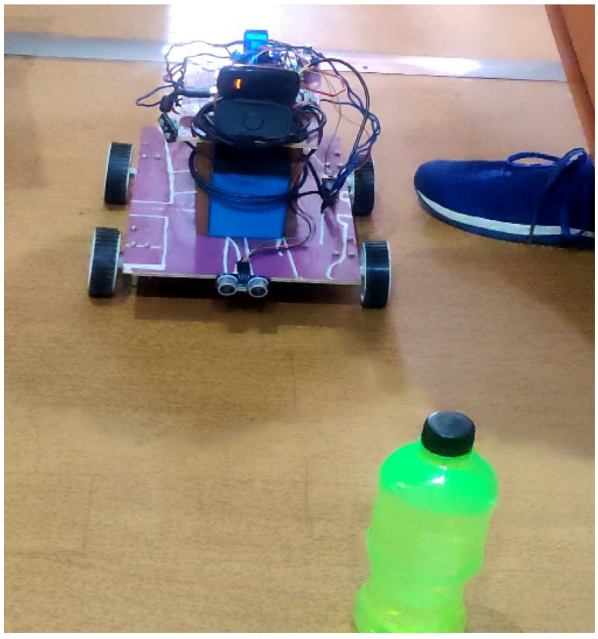

Fig. 12. Testing Object detection (b).

constant eye on the respective object in real-time. Also, the robot can be controlled manually through user interface. To see the working video taken during testing the robot whose images are attached above can be seen at https://ai-robot.herokuapp.com/.

Fig. 13. Testing Object detection (c).

4. Conclusion

With the current ongoing research in computer vision, Object following will always be an evergreen research topic with multiple daily life applications. An object detection and following robot is proposed which can be used in many applications which can help human being in diverse fields which include object following with user control.

CRediT authorship contribution statement

Anirban C. Mitra: Supervision. **Anand Kamble:** Methodology, Software. **Aniket Tathe:** Conceptualization, Software, Writing – original draft, Writing – review & editing. **Suyash Kumbharkar:** Formal analysis, Validation, Methodology. **Atharva Bhandare:** Data curation, Resources.

Data availability

Data will be made available on request.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We would like to thank our college MESCOE for providing us all the facilities to do this project.

References

[1] R. Girshick, J. Donahue, T. Darrell, J. Malik, Region-Based Convolutional Networks for Accurate Object Detection and Segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 38 (1) (2016) 142–158.
[2] R. Girshick, Fast R-CNN, in: IEEE International Conference on Computer Vision (ICCV) 2015, 2015, pp. 1440–1448, https://doi.org/10.1109/ICCV.2015.169.
[3] Liu, Wei & Anguelov, Dragomir & Erhan, Dumitru & Szegedy, Christian & Reed, Scott & Fu, Cheng-Yang & Berg, Alexander. (2016). SSD: Single Shot MultiBox Detector. 9905. 21-37. 10.1007/978-3-319-46448-0_2.
[4] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You Only Look Once: Unified, Real-Time Object Detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016, 2016, pp. 779–788, https://doi.org/10.1109/CVPR.2016.91.
[6] Lin TY. et al. (2014) Microsoft COCO: Common Objects in Context. In: Fleet D., Pajdla T., Schiele B., Tuytelaars T. (eds) Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, vol 8693. Springer, Cham.
[8] Maier, Alexander & Sharp, Andrew & Vagapov, Yuriy. (2017). Comparative Analysis and Practical Implementation of the ESP32 Microcontroller Module for the Internet of Things. 10.1109/ITECHA.2017.8101926.
[9] A. Howard et al., Searching for MobileNetV3, in: IEEE/CVF International Conference on Computer Vision (ICCV) 2019, 2019, pp. 1314–1324, https://doi.org/10.1109/ICCV.2019.00140.
[10] V. Pimentel, B.G. Nickerson, Communicating and Displaying Real-Time Data with WebSocket, IEEE Internet Comput. 16 (4) (2012) 45–53.

Further Reading

[5] F. Najafi, P. Wang, D. O. Nyabuga, "HybridEI: Smartly Face Detection System in Resource Constrained Edge Environment," 2021 IEEE International Conference on Networking, Sensing and Control (ICNSC), 2021, pp. 1-6, doi: 10.1109/ICNSC52481.2021.9702180.
[7] H. Sun, P. Gong, A Safety-Helmet Detection Algorithm Based on Attention Mechanism, in: 2021 7th IEEE International Conference on Network Intelligence and Digital Content (IC-NIDC), 2021, pp. 11–15, https://doi.org/10.1109/IC-NIDC54101.2021.9660439.
[11] Y. Tu, M. Urata, M. Endo, T. Yasuda, H. Shimazaki, T. Kimura, Development of Land Use Judgment System using Deep Learning to Support Land Evaluation, in: 2021 IEEE 10th Global Conference on Consumer Electronics (GCCE), 2021, pp. 332–336, https://doi.org/10.1109/GCCE53005.2021.9621770.
[12] B. Fu, S. Li, J. Wei, Q. Li, Q. Wang, J. Tu, A Novel Intelligent Garbage Classification System Based on Deep Learning and an Embedded Linux System, IEEE Access 9 (2021) 131134–131146, https://doi.org/10.1109/ACCESS.2021.3114496.
[13] F. Yu, W. Li, Q. Li, Y. Liu, X. Shi, J. Yan, Poi: Multiple object tracking with high performance detection and appearance feature, in: ECCV, Springer, 2016, pp. 36–42.
[14] L. Li and L. Zhou, Real-Time Facial Landmark Detection by Attention-driven Lightweight Network, 2021 IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), 2021, pp. 290–294, doi: 10.1109/IMCEC51613.2021.9482202.
[15] A. Bewley, Z. Ge, L. Ott, F. Ramos, B. Upcroft, Simple online and realtime tracking, in: 2016 IEEE International Conference on Image Processing (ICIP), IEEE, 2016, pp. 3464–3468.