Deep Learning Obstacle Detection and Avoidance for Powered Wheelchair

Yahya Tawil
Computer Engineering Department
Hasan Kalyoncu University
Gaziantep, Turkey
yahya.tawil@std.hku.edu.tr

*A.H. Abdul Hafez**
Computer Engineering Department
Hasan Kalyoncu University
Gaziantep, Turkey
abdul.hafez@hku.edu.tr

***Abstract*—Depth sensors like RGB-D cameras, LiDARs and laser scanners are widely investigated in research for Smart Wheelchair (SW) to carry out navigation, localization and obstacle detection and avoidance tasks. These sensors are costly compared to monocular camera sensor. A single off-the-shelf camera can be an economically efficient sensor to achieve obstacle detection and avoidance. We present in this paper a single camera based obstacle detection and avoidance method without using any 3D information. It is a novel vision-only system for wheelchair obstacle detection and avoidance that uses a Raspberry Pi along with Raspberry Pi camera. The obstacles are detected using a deep learning model built on MobileNetV2 SSD. The model is retrained using a dedicated dataset that was built for this purpose. Bounding boxes are used to mark detected obstacles; and feed them as features to the image space obstacle avoidance module. Figure 1 depicts internal view of what does the system see and an abstract description of our system's functionality.**

***Keywords*—Autonomous wheelchair, obstacle detection and avoidance, Assistive technology, Deep Learning, Computer Vision.**

I. INTRODUCTION

Obstacle avoidance is one of the challenges that wheelchair users with certain types of paralysis or other disabilities. The risk of falling, collision or injuries is high in such cases. Through this work, we demonstrate that providing independence using autonomous smart wheelchair is achievable thanks to advanced technologies in computer vision and deep learning. Our work aims to increase the mobility of people with disabilities in outdoor environments. Through the use of a single low-cost camera sensor with a wheelchair, our work aims to reduce the costs of using expensive sensors like LiDAR and RGB-Depth camera. This will allow access by numerous disabled users with special mobility needs.

For robots using the sidewalk, like wheelchairs, there is a lack of datasets to enable deep learning model training, motivating us to initiate the building of our own essential dataset. The obstacle detection is achieved by retraining the MobileNetV2 SSD network using the dataset that we developed for the purpose. The ability of the network to detect objects in the image has enabled us to develop an image space obstacle avoidance, which uses an image-based visual servoing control law. The system developed involves a low-cost setup consisting of Raspberry Pi 4 and a Raspberry camera.

*Corresponding author

978-1-6654-8894-5/22/$31.00 ©2022 IEEE

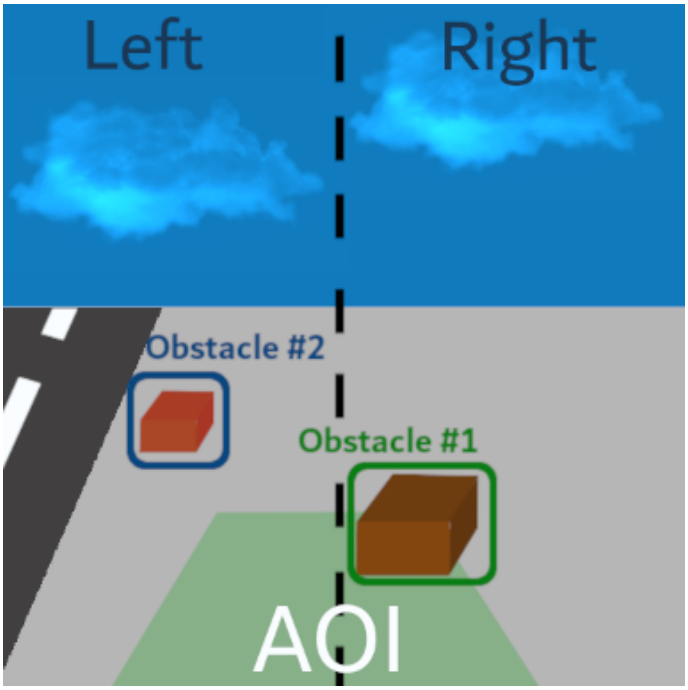

Fig 1: The proposed system works on pushing the detected obstacle inside the Area Of Interest (AOI) to the safety margin of the image.

Wheelchairs obstacle avoidance was the topic of work by several researchers. For example, in [1] an artificial potential field was used to avoid obstacles detected by a redundant number of ultrasonic sensors, which proved to be less reliable and accurate than the camera sensor. In [2] depth information extracted from a Kinect sensor was used to calculate the forces of the potential field. The Kinect sensor is costly alternative to the Raspberry Pi Camera used in our work, in addition to which they used Cartesian space information, rather than the image space that we adopted. A few wheelchair studies have been done using classical computer vision algorithms [3], [4], and currently outperformed by deep learning approach.

The work presented in [3] uses a combination of multiple ultrasonic sensors surrounding the wheelchair body and a camera to implement functionality in respect of obstacle detection and avoidance, able to recognize two critical environmental situations in front of the chair. It is considered a driving assistance system that still requires user action.

2022 Innovations in Intelligent Systems and Applications Conference (ASYU) | 978-1-6654-8894-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ASYU56188.2022.9925493

TABLE I: The different datasets built for or used in object detection for robots that use side-walk environments.

Dataset	Size	Annotation	Classes
SideGuide [11]	350K	Bounding box & Semantic polygon	29
SOID [13]	50K	No annotations (classification)	5
Vistas [15]	25K	Semantic polygon	37
PESID [14]	1.9K	N/A* (seems bounding box)	10
CDBV [12]	13.5K	N/A* (seems bounding box)	6

* We don't have access to the dataset.

Previous works considered the indoor environment [5], [6] focusing on deep learning and Gaussian processes for corridor navigation. Wheelchair avoidance, localization and negotiation of doorways is considered in [7] and [8], where a dynamic window method is used for avoidance and adaptive Monte Carlo for localization. This work used costly RGB-D camera and was for indoor environment.

Other works used deep learning in the smart wheelchair systems. A CNN-based system is used in [9] to help wheelchairs to board buses by detecting the bus and its door open using YOLO and a LiDAR to provide measurements required to pass through the door. The work captured in [10] suggests a set of steering actions to wheelchair user as an assisting tool rather than autonomous driving, and uses a deep assessing neural network based on Long Short Term Memory (LSTM) neural network takes distance information input from three ultrasonic to sense the surrounding obstacles and suggest a correct direction in response.

The research community working on autonomous vehicles has produced several datasets. However, only few of them and with small size are suitable for the autonomous wheelchair in general and for travel on sidewalks in particular. We summarize in the table I the available datasets that can be used for robots operating in sidewalk environments. All the datasets mentioned in the table, other than Vistas, have accessibility issues, which is what led us to build our own dataset with open access policy.

It can be concluded from the above review of the previous works that most use costly non-vision systems, and only few employ deep learning algorithms, but with limited applications. By contrast, our paper, to the best of our knowledge, is the first to contribute and integrate the following to the wheelchair system: i) deep learning obstacle detection for a wheelchair system deployed using Raspberry Pi; ii) image space wheelchair obstacle avoidance control law.

Our paper is organized as follows. Section 2 describes the deep learning model we have chosen for object detection and the dataset we have built for fine-tuning the model. Section 3 introduces The control law designed for obstacle avoidance using image space. Sections 4 presents the experiments done to evaluate the accuracy of obstacle detection model, then the deployment of our system using a real power wheelchair and assessing the obstacle avoidance control law in real scenarios inside our university campus. Finally, Section 5 presents our conclusions.

II. The Deep Model and Dataset for Retraining

This section describes the deep learning network and the dataset we developed and used in the retraining process.

TABLE II: Number of instances per class in our dataset.

Traffic sign	Trunk	Trash can	Person	Rock	Fire hydrant
1415	884	317	165	164	116

We have selected MobileNetV2 SSD (SSDLiteM2) model for this task. SSDLiteM2 was chosen due to its satisfactory performance and accuracy on Raspberry Pi 4, our adopted hardware, as well as to its availability as a pre-trained model within the TensorFlow development platforms.

SSDLiteM2 takes RGB image with a 320x320 pixel size as input and returns the positions of the detected boxes in the output, along with its prediction percentage value. SSDLiteM2 consists of Mobilenetv2 [16] as the backbone model for feature extraction and Single Shot multibox Detector (SSDLite) [17] as the head model. MobileNet is a commonly used CNN architecture as an image feature extractor for detection and classification, originally designed to run on mobiles and embedded hardware. Whereas SSD is a one-stage detector. SSD uses a group of small convolutional filters in the bounding box predictor to predict category scores and box offsets for a fixed set of default bounding boxes.

The MobileNetV2 SSD is pre-trained using the COCO 2017 dataset with 80 classes. We defined 6 obstacle classes in our dataset to be detected and yet the model is customized to have only 6 classes. Only the box regression and classification heads of the detection (SSD) model were retrained without the feature extraction part in MobileNetv2. The ratio of training to testing sets is 90:10 in our dataset.

A total number of 835 were collected in our dataset and distributed over six different classes annotated in the format of bounding boxes using Edge Impulse platform annotation tool. The six classes and distribution of data over classes is depicted in Table II. We added various types of noise and other visual effects to the images to replicate the different lighting that might be encountered in an outdoor environment doubling the size of the original dataset. Figure 2 provides a set of images showing the applied augmentation: quality adjustment, brightness correction, Gaussian noise addition, and contrast.

III. The Proposed Image Space Wheelchair Obstacle Detection and Avoidance

This section presents the proposed obstacle avoidance method, an image space vision-guided avoidance. The overall process is captured in an illustration diagram in Figure 3. The image acquired by the vision sensor is fed as an input to the obstacle detection module. The output of this module is a bounding box that defines the location of the obstacle in the image. The obstacle coordinates are sent to the avoidance module to take the required avoidance action.

To simplify the presentation, we assumed that the wheelchair was following the sidewalk in a straight line motion through the main task and will resume the main task as soon as the avoidance task is completed. The design of the control law that achieves this motion is based on an Image Based Visual Servoing (IBVS) controller.

Fig 2: Sample images from the dataset showing the four augmentation effects.

The problem formulation in the image space is illustrated in Figure 4. We selected the coordinates of the center of the bounding box around the detected obstacle as a feature. We assume that only one obstacle would appear in the AOI at any one time. The size of the image is $I_c \times I_r$. If the obstacle is in the left half of the image then it will be moved to the left side, and will be moved to the right side if it is in the right half of the image. The current feature point position of the bounding box around the obstacle is $s = (u, v)$, and are aimed to be moved to the desired feature point position $s^* = (u^*, v^*)$. As the aim of the avoidance process to move the obstacle to the right or to the left outside the image, the coordinate v^* is selected to be the same value of v. The coordinate u^* is selected in the middle of the margin depending on whether the obstacle is in the right or left half

$$u^* = \begin{cases} I_c - \frac{m}{2} & : \text{ feature point} \in \text{right half} \\ \frac{m}{2} & : \text{ feature point} \in \text{left half} \end{cases} \quad (1)$$

Here, m is the safety margin width in pixels.

We define the task function $e = (s - s^*)$ to be regulated to zero, and let $\dot{r} = (\nu_y, \omega_z)^T$ represents the velocity to move the wheelchair. It is composed of linear velocity along Y-axis (ν_y) and angular velocity around Z-axis (ω_z).

we can relate the velocity of feature point to the wheelchair velocity in Cartesian frame as:

$$\dot{s} = J\dot{r} \quad (2)$$

By substitution the image Jacobian matrix J in (2):

$$\begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} -v\omega_z \\ \frac{\lambda \nu_v}{Z} + u\omega_z \end{pmatrix} \quad (3)$$

Where, u and v are the image point coordinates, and Z is the depth of the 3D point. Also, λ is the camera focal length in pixels. For a simple control law design, we assume an exponential convergence of the error so $\dot{e} = -\alpha e$, or $\dot{u} = -\alpha(u - u^*)$, and $\dot{v} = -\alpha(v - v^*)$. In the ideal case $v - v^* = 0$. Z is considered constant and equal to $Z = 1.5$ meter. Thus substituting in (3) produces:

$$\begin{aligned} \omega_z &= \frac{(u - u^*)}{v} \\ \nu_y &= -\alpha \frac{u(u - u^*)}{v} \frac{Z}{\lambda} \end{aligned} \quad (4)$$

Here, α is a gain constant, The above calculated values of ω_z and ν_y ensure to regulate the error e to zero. This means the center of the obstacle is moved to the safety margin in the image by moving the wheelchair away from the obstacle.

The wheelchair is modelled as a six wheels robot moving on a horizontal plane. The two large rear wheels are actuated by two DC motors. The other four wheels are passive, and used to support the wheelchair. The wheelchair and camera configuration is depicted in Figure 5. The world coordinate frame is defined as F_w and the wheelchair frame (robot frame) is defined as F_r. This frame is attached to the middle of the segment formed by the centers of the two deferentially actuated wheels. The camera frame is defined as F_c. These frames are similarly defined in [18].

IV. EXPERIMENTS

This section presents our experimental setup and our experimental evaluation of the proposed obstacle detection and avoidance. Our developed system runs on Raspberry Pi 4 with a Raspberry Pi camera v2.1 module. We used a Sabertooth driver to control the wheelchair motors and experimented with the wheelchair around our university campus.

The velocity of the unicycle robot shown in Figure 5 in the global coordinate frame where the control inputs $u = (\nu, \omega)$ is given by the following [20]: $v_x = v\cos\varphi$, $v_y = \omega \sin\varphi$, $\omega_z = \omega$

In our obstacle avoidance task, we controlled the motion of the wheelchair during the avoidance using the linear velocity along Y-axis ν_y, and the angular velocity around Z-axis ω_z. These two are calculated finally in the world coordinate frame F_w. To simplify the design of the low-level controller, we considered normalized values for the velocities ν_y and ω_z. These two values are scaled using a manually tuned scale parameters σ_ν and σ_ω.

A. Training and evaluating the obstacle detection model

The object detection in this work built on top of MobileNetV2 SSD FPN-Lite 320x320 available in TensorFlow detection models repository pretrained.

We used during building the dataset a tool called Feature Explorer presented in Figure 6. This technique is based on Uniform Manifold Approximation and Projection (UMAP) [19]

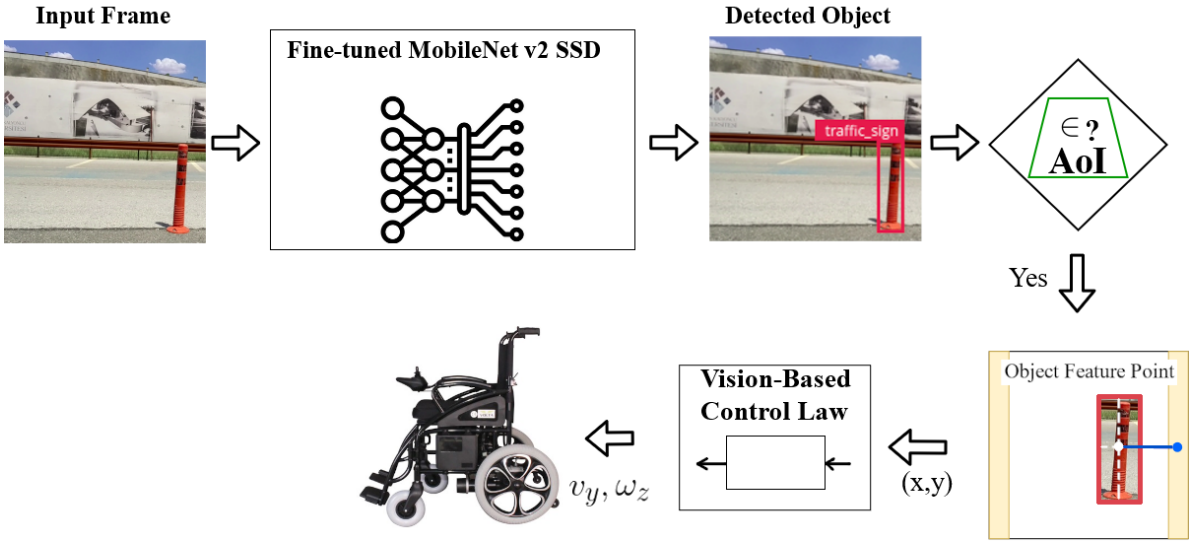

Fig 3: Employing the trained object detection model in our smart wheelchair obstacle detection and avoidance system.

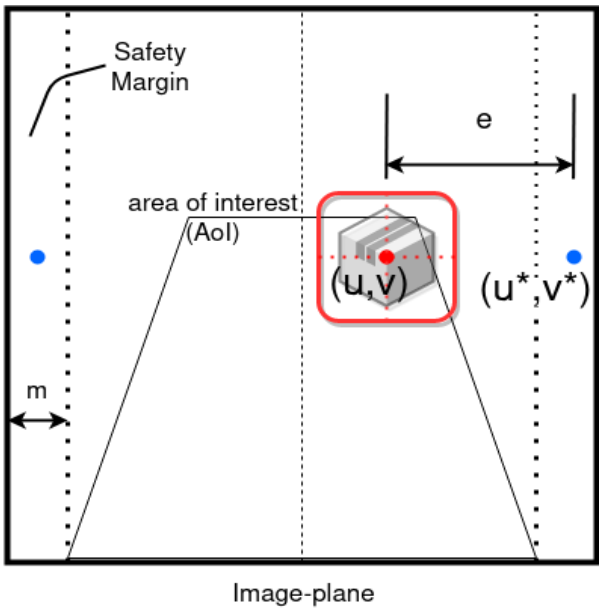

Fig 4: Demonstrating the problem in the image plane. It depicted the: AOI, safety margin, error function e , position of feature point (u, v), and the position of desired location of feature point (u^*, v^*).

for data dimensionality reduction. As object detection task is based on image classification techniques, the Feature Explorer helped us to detect outliers in the dataset by visualizing how good the samples can be classified. Additional enhancements based on feature explorer will be considered in a next dataset version.

Our experiments have reported 61% mAP accuracy after training the model with total of 835 training images available in our dataset. However, a 66% mAP for MobileNetv2 SSD is reported in [21] after training the model with PASCAL VOC dataset that contains 20 classes. This gives some insights into our resultant mAP. In Figure 7, we present samples from the resultant model validation set images.

TABLE III: Evolution of mAP results during fine-tuning experiments.

Trail	Remark	Images#	Epochs	Instances#	mAP
#1	Initiate	645	25	2422	27.4
#2	Increase images number	853	35	3061	52.8
#3	with_augmentation	853 + 1706	30	3061	60.1

Table III presents performance analysis of the retrained model while increasing the number of training images. The effect of adding visual effects to the training images is noticeable in the last row of Table III, where the mAP value has been boosted from 27% to 61%.

By deploying using Raspberry Pi, we achieve a frame rate of between 0.8 and 1.6 (fps). The maximum frame rate was obtained by increasing the Raspberry Pi CPU clock to 900 Mhz and increasing the Python Interpreter Process priority by the OS.

B. Experiments on the detection and avoidance using the real wheelchair system

Multiple experiments with different obstacles in different scenarios were conducted in our university campus. We selected one of these experiments in Figure 8 showing the output of the control low while avoiding one obstacle.

In Figure 8, images (a) and (b) shows from an external view the wheelchair detecting the traffic sign inside the AOI, the control law computed the positive velocity to push the obstacle feature point to the safety margin by moving the wheelchair to the right. However, the required velocity in (a) is higher than (b) as the obstacle is closer to safety margin in (b). (c)-(d) are identical to (a)-(b) but from the internal system view.

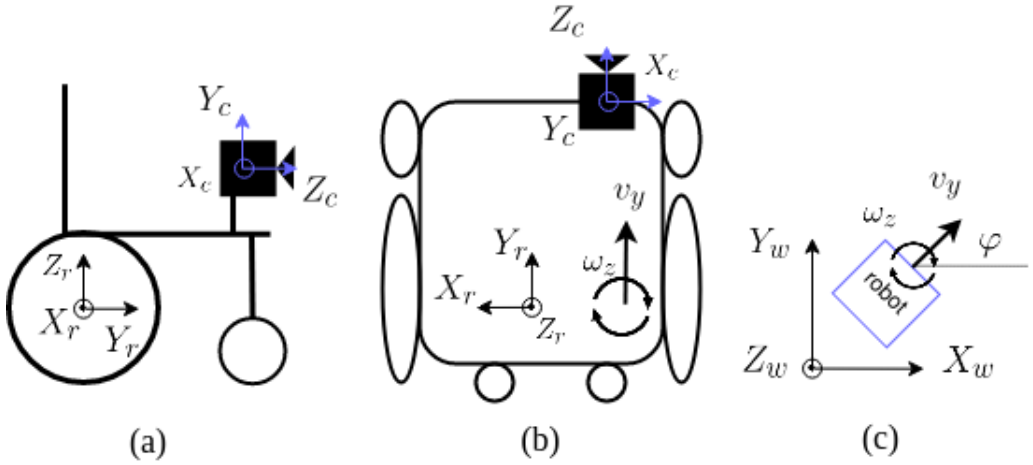

Fig 5: Wheelchair modelling with camera, robot and world frames (a) Side view of the wheelchair (b) Top view of the wheelchair (b) The wheelchair and world frame.

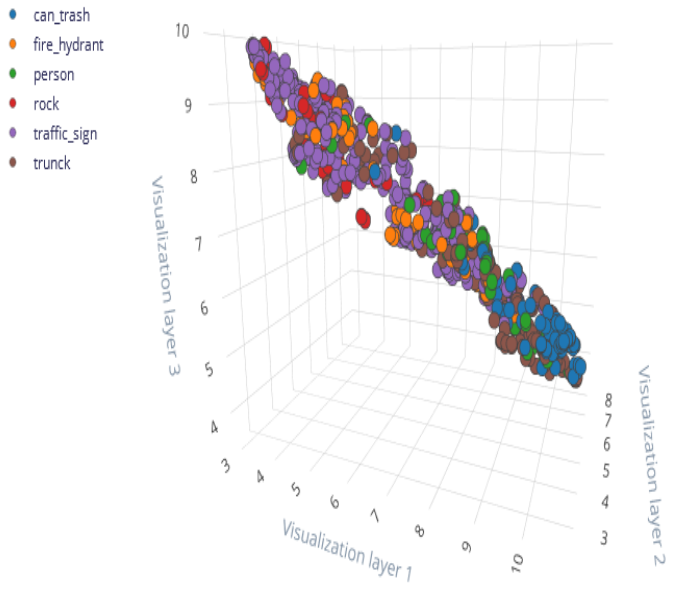

Fig 6: Our dataset visualized using feature points after dimensionality reduction to 3D space. It shows how each class presents a cluster, nearly.

Fig 7: First row are the validation samples images with bounding boxes as ground truth and the second row are output from the object detection sub-system after fine-tuning with the pilot dataset with prediction confidence.

The wheelchair is initially moving straight with a speed corresponding to the main task requirements, we assume that the main task requirements is to move the wheelchair in a straight line with constant velocity.

V. Conclusions and Discussion

We have built our own dataset for a smart wheelchair traveling on sidewalks. This bridges the gap of datasets for autonomous wheelchair navigation in outdoor environments. We showed how our dataset can be used effectively to build a low-cost wheelchair obstacle detection and avoidance system by retraining an object detection model using it. Our results of achieving 61% mAP for obstacles detection and having control law capable to avoid the obstacle effectively are promising and address one of the key challenges wheelchair users face during outdoor mobility.

As a future work, we are looking to increase the size of the dataset in terms of the images and the number of classes, and additionally performing more experiments with Nvidia Jetson Nano and optimizing the deep learning model to achieve better speed. Moreover, we are looking forward integrating a path planning algorithm to give the wheelchair the ability to arrive at a target place while avoiding obstacles in its route.

Acknowledgment

This work is supported by TUBITAK under project number 117E173.

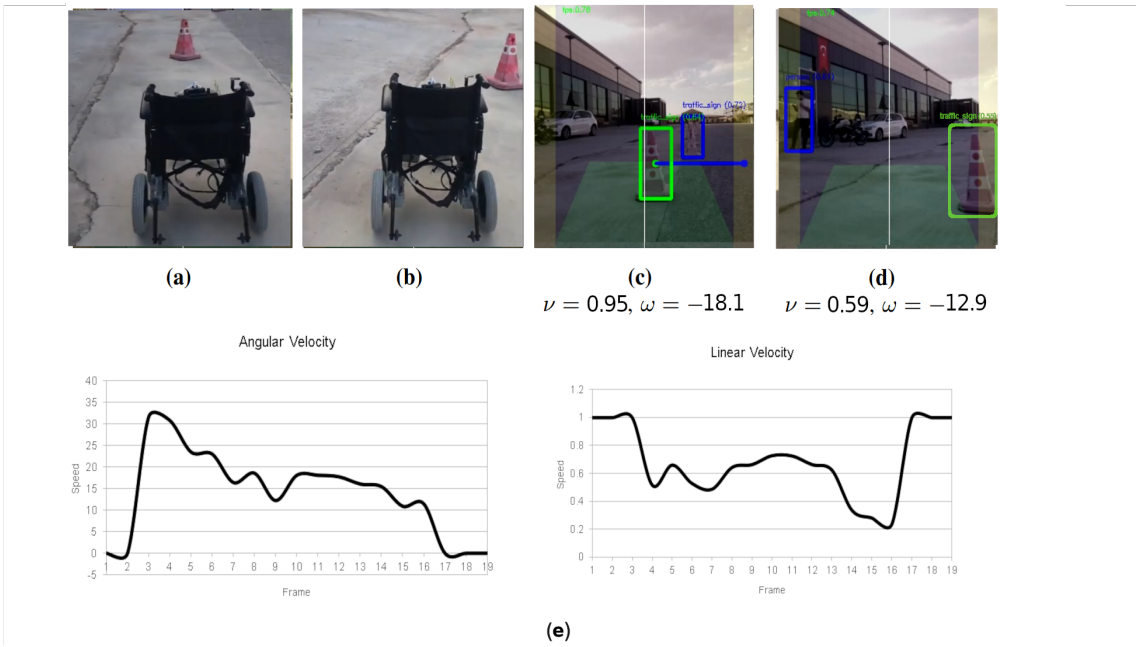

Fig 8: Two different scenes in (a)-(b) from external view for an obstacle detected inside the AOI in (a) and avoided in (b). (c)-(d) shows the system view of the detected obstacle (green bounding box) inside AOI (green trapezoid area) and how control law is bringing the feature point (green dot) to the safety margin based on distance to it (blue line). Obstacles outside the AOI are in blue bounding box. Under each image the action generated using the velocities calculated in the control law. (e) shows the angular and linear speed calculated by the control law while avoiding the obstacle in the scenes (c)-(d).

REFERENCES

[1] Diao, C, Jia, S, Zhang, G, Sun, Y, Zhang, X, Xue, Y, Li, X. "Design and realization of a novel obstacle avoidance algorithm for intelligent wheelchair bed using ultrasonic sensors". In 2017 Chinese Automation Congress (CAC) 2017 (pp. 4153–4158).

[2] Özkörs, M, , A, Say, M, Yazici, A, Yayan, U, Ak, M. "Kinect based Intelligent Wheelchair navigation with potential fields". In 2014 IEEE International Symposium on Innovations in Intelligent Systems and Applications (INISTA) Proceedings 2014 (pp. 330–337).

[3] Kim, E. "Wheelchair navigation system for disabled and elderly people". Sensors 2016; 16(11):1806.

[4] Lee, Y, Lim, J, Eu, K, Goh, Y, Tew, Y. "Real time image processing based obstacle avoidance and navigation system for autonomous wheelchair application". In 2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC) 2017 (pp. 380–385).

[5] Dorbala, V, Hafez, A, Jawahar, CA . "Deep Learning Approach for Robust Corridor Following". In 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2019 (pp. 3712–3718).

[6] Tello, A, Hafez, A, Sarakbi, B. "Real-time GP-based Wheelchair Corridor Following". In 2021 29th Signal Processing and Communications Applications Conference (SIU) 2021 (pp. 1–4).

[7] Li, Z, Xiong, Y, Zhou, L. "ROS-based indoor autonomous exploration and navigation wheelchair". In 2017 10th International Symposium on Computational Intelligence and Design (ISCID) 2017 (pp. 132–135).

[8] Burhanpurkar, M, Labbé, M, Guan, C, Michaud, F, Kelly, J. "Cheap or robust? the practical realization of self-driving wheelchair technology". In 2017 International Conference on Rehabilitation Robotics (ICORR) 2017 (pp. 1079–1086).

[9] Ali, S, Al Mamun, S, Fukuda, H, Lam, A, Kobayashi, Y, Kuno, Y. "Smart robotic wheelchair for bus boarding using CNN combined with hough transforms". In International Conference on Intelligent Computing 2018 (pp. 163–172).

[10] Haddad, M, Sanders, D. "Deep learning architecture to assist with steering a powered wheelchair". IEEE Transactions on Neural Systems and Rehabilitation Engineering 2020; 28(12):2987–2994.

[11] Park, K, Oh, Y, Ham, S, Joo, K, Kim, H, Kum, H, Kweon, I. "SideGuide: A Large-scale Sidewalk Dataset for Guiding Impaired People". In 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2020 (pp. 10022–10029).

[12] He, Y, Yang, HB, Wang, SJ. "CDBV: A Driving Dataset With Chinese Characteristics From a Bike View". IEEE Access 2019; 7:51714–51723.

[13] Ahmed, F, Yeasin, M. "Optimization and evaluation of deep architectures for ambient awareness on a sidewalk". In 2017 International Joint Conference on Neural Networks (IJCNN) 2017 (pp. 2692–2697).

[14] Sun, C, Su, J, Ren, W, Guan, Y. "Wide-View Sidewalk Dataset Based Pedestrian Safety Application". IEEE Access 2019; 7:151399–151408.

[15] Neuhold, G, Ollmann, T, Rota Bulo, S, Kontschieder, P. "The mapillary vistas dataset for semantic understanding of street scenes". In Proceedings of the IEEE international conference on computer vision 2017 (pp. 4990–4999).

[16] Sandler, M, Howard, A, Zhu, M, Zhmoginov, A, Chen, LC. "Mobilenetv2: Inverted residuals and linear bottlenecks". In Proceedings of the IEEE conference on computer vision and pattern recognition 2018 (pp. 4510–4520).

[17] Liu, W, Anguelov, D, Erhan, D, Szegedy, C, Reed, S, Fu, CY, Berg, A. "Ssd: Single shot multibox detector". In European conference on computer vision 2016 (pp. 21–37).

[18] Pasteau, F, Narayanan, V, Babel, M, Chaumette, F. "A visual servoing approach for autonomous corridor following and doorway passing in a wheelchair". Robotics and Autonomous Systems 2016; 75:28–40.

[19] McInnes, L, Healy, J, Melville, J. "Umap: Uniform manifold approximation and projection for dimension reduction". arXiv preprint arXiv:1802.03426 2018.

[20] Gulati, S, Kuipers, B. "High performance control for graceful motion of an intelligent wheelchair". In 2008 IEEE International Conference on Robotics and Automation 2008 (pp. 3932–3938).

[21] Pan, J, Sun, H, Song, Z, Han, J. "Dual-Resolution Dual-Path Convolutional Neural Networks for Fast Object Detection". Sensors 2019; 19(14):3111.