Controlling Autonomous Vehicles in Pedestrian Spaces Using Neural Networks: A Study on Model Complexity

Reza Rafeh
Department of Information Technology
Crown Instittute of Higher Education
Canberra, Australia
reza.rafeh@cihe.edu.au

Vahid Rafe
Center for Software Reliability
University of London
St George's, UK

Tyler Barham
Department of Information Technology
Waikato Institute of Technology,
Hamilton, New Zealand

***Abstract*— The increasing presence of autonomous and semi-autonomous vehicles in pedestrian spaces, such as rent-to-ride e-scooters, has raised important safety and operational challenges. This study explores the use of neural networks (NNs) for vehicle navigation and control in such environments, using camera images and GPS data as inputs. Specifically, we examine how varying the size of convolutional neural networks (CNNs) influences both classification accuracy and the practical feasibility of real-time deployment.**

A set of CNN models, inspired by the AlexNet architecture, were trained on a dataset of vehicle-mounted camera images and corresponding driving actions (e.g., accelerate, brake, turn). The evaluation focused on classification performance using AUROC metrics and observed runtime behavior in a simulated environment. While larger models demonstrated stronger predictive accuracy, only smaller networks were capable of real-time operation due to hardware constraints.

These findings highlight the trade-off between network complexity and deployability in pedestrian-focused autonomous systems. Additionally, the study underscores concerns around the over-reliance on GPS data and the limitations of vision-only approaches in unstructured environments. Further work is needed to strengthen decision-making robustness and support safe, effective AV deployment in shared pedestrian spaces.

Keywords— Autonomous Vehicle, Micro-mobility, Convolutional Neural Network, AlexNet, Collision Avoidance, Model Complexity.

I. INTRODUCTION

Autonomous vehicles (AVs) have seen rapid development over the past decade, with major advancements in computer vision, deep learning, and robotics enabling the creation of intelligent systems capable of navigating complex environments. However, while road-based AVs dominate mainstream development, a subset of micro-mobility AVs that operate on pedestrian pathways or sidewalks has begun to emerge. These footpath-occupying AVs have a range of practical applications, including lastmile delivery, campus transport services, accessibility support for people with limited mobility, and automated repositioning of rent-to-ride e-scooters [1].

Footpath-occupying AVs present unique challenges. These platforms must be compact, power-efficient, and capable of making split-second decisions in crowded, unpredictable pedestrian spaces. Unlike road-based vehicles, footpath AVs often lack the computational resources to deploy large-scale deep learning models. This makes model size, inference time, and energy efficiency critical performance factors [2].

This study focuses on how the model complexity, in particular, architectural depth of neural networks affects the AV's performance in such constrained environments. We hypothesize that moderately deep networks may offer a balance between prediction accuracy and system responsiveness, making them more suitable for deployment in real-world sidewalk navigation scenarios.

The remaining sections of this paper are organized as follows. Section 2 reviews related research projects. Section 3 outlines the research methodology. Section 4 describes the training of neural networks and the setup of simulation environments for evaluation. Finally, Section 5 concludes the paper and proposes future directions for the research.

II. RELATED WORK

Autonomous vehicle navigation has traditionally focused on structured environments such as highways and urban roads, where predictable traffic patterns and defined lanes allow for rule-based systems and high-definition maps to aid navigation. However, sidewalk or footpath-based navigation introduces a different set of challenges, including unstructured environments, unpredictable pedestrian movement, and dynamic obstacles such as cyclists, pets, and street furniture.

Deep learning has emerged as a powerful tool for perception and decision-making in autonomous systems. Research has demonstrated the feasibility of end-to-end learning using convolutional neural networks (CNNs) to map raw pixels to steering commands in road environments [3]. However, these models often require substantial computational resources and are designed for environments where real-time constraints are met with high-performance hardware.

MobileNet, SqueezeNet, and other lightweight architectures have been developed to address this limitation. These models reduce the number of parameters through architectural innovations such as depthwise separable convolutions and bottleneck layers, allowing deployment on embedded systems like mobile phones and IoT devices. MobileNetV2 was shown to maintain competitive accuracy with significantly lower computational overhead, making it suitable for applications like footpath AVs [4]. Despite these advances, little work has specifically addressed the trade-off between model size and performance in constrained sidewalk AV scenarios. Studies have explored pruning and quantization techniques to reduce model size post-training, but these often come with a compromise in accuracy or require fine-tuning to preserve performance [5]. Furthermore, most research in model optimization assumes a powerful training phase with subsequent model shrinking, whereas in sidewalk AV

applications, even real-time inference constraints on consumer-grade hardware are a primary concern [6]. Some research has investigated the effects of model depth and parameter count on decision latency. For example, [7] explored how deep compression techniques could allow large models to fit on-chip memory and improve runtime efficiency. However, their focus was on post-hoc compression, not architecture selection.

Similarly, edge computing research has considered deploying CNNs on Jetson Nano or Raspberry Pi platforms, but these studies often focus on object detection rather than end-to-end navigation tasks [8].

In simulation-based AV training, platforms such as CARLA and AirSim have been popular [9]. However, this study uniquely uses a modded version of Grand Theft Auto V (GTA V) to simulate footpath navigation, offering a visually realistic and dynamic testbed. While not as controllable as academic simulators, GTA V provides rich urban environments and pedestrian behavior that more closely reflect real-world sidewalk scenarios [10].

Recent studies have explored the impact of varying convolutional filter sizes and numbers in CNNs for human motion classification. Optimal configurations achieved up to 98.98% accuracy, highlighting the importance of adaptive filter design across layers [11].

This work contributes to the field by evaluating the impact of neural network size on autonomous vehicle (AV) performance, specifically within footpath environments that include dynamic obstacles such as pedestrians and cyclists, as well as static elements like trees and rubbish bins, contexts where swift and accurate control is essential. It fills a gap in the literature by investigating how varying CNN depths influence both prediction accuracy and inference speed in a realistic, dynamic simulation setting. Unlike prior studies that focus solely on accuracy or object detection, this research addresses practical deployment constraints and prioritizes human safety in real-time navigation.

III. Research Methodology

This section outlines the methodological framework used to investigate how the number of CNN layers influences the performance of footpath AVs in a simulated environment. Grounded in Information Processing Theory (IPT), the research explores the relationship between network architecture (analogous to long-term memory) and the autonomous vehicle's decision-making output (response). To simulate real-world pedestrian environments safely and reproducibly, the experiments were conducted using a virtual city environment within Grand Theft Auto V.

Three convolutional neural network (CNN) structures inspired by AlexNet were designed with varying complexities, and each was trained over multiple durations to assess generalisation performance. The methodology includes the design of the neural networks, data collection procedures, model training and evaluation using AUROC metrics, as well as a consideration of computational constraints. This approach enabled a controlled investigation into how neural network depth and parameter count affect real-time navigation performance in a constrained computing environment.

A. Experimental Design

The experimental setup was designed to isolate the effect of neural network size and training exposure on AV performance. To do this, we implemented three CNN structures inspired by AlexNet which are shown in Table I Reference source not found., each differing in complexity.

TABLE I. Test cases and their independent variables

Epochs	**Structure 1 Layer width = n Trainable parameters = 70,881,860**	**Structure 2 Layer width = n/2 Trainable parameters = 17,735,972**	**Structure 3 Layer width = n/4, Trainable parameters = 4,441,748**
15	Test case 1	Test case 4	Test case 7
30	Test case 2	Test case 5	Test case 8
60	Test case 3	Test case 6	Test case 9

B. Dataset Preparation

To train and test the CNNs, a dataset of labeled images was gathered using the Grand Theft Auto V simulation environment. We manually drove an in-game vehicle along footpaths while collecting camera images paired with corresponding user inputs (e.g., accelerate, brake, turn left, turn right). The camera view also included a mini-map overlay, simulating a GPS region input. Fig. 1 shows the camera view from the AV.

Fig. 1. The view of AV's camera

A total of 18,000 image-action pairs were collected for training, with an additional 3,000 samples used for testing. Environmental variables, such as time of day, weather, and pedestrian density, were held constant during data collection to ensure consistency and reduce noise.

C. Neural Network Configuration and Training

The three CNN architectures were adapted from AlexNet but modified for the experimentations. Specifically:

- The input layer accepted compressed RGB images of 256×192 pixels.
- The output layer consisted of four sigmoid-activated values, each representing the confidence for a vehicle action (e.g., turn left, brake). Multiple actions could be predicted simultaneously.

Each of the nine test cases (3 structures × 3 epochs) was trained using identical preprocessing and hyperparameters. Training was performed in the Kaggle cloud environment using a Tesla P100 GPU, enabling faster training for larger networks that could not be processed locally due to hardware limitations.

D. *Evaluation Metrics*

To evaluate model performance, the study used the Area Under the Receiver Operating Characteristic (AUROC) curve. This metric quantifies the model's ability to distinguish between classes across different thresholds. AUROC was chosen for its effectiveness in multi-class classification settings, especially where class imbalance may be present. Performance was assessed separately for each of the four action outputs, and per-class

AUROC scores were averaged to compare results across models and test cases. The AUROC benchmark of 0.5 (equivalent to random guessing) was used to interpret significance.

IV. MODEL Training and Performance Analysis

This section presents the analysis conducted to prepare, validate, and interpret the performance of neural networks designed to guide autonomous vehicles (AVs) in pedestrian spaces. The analysis process began with determining appropriate training strategies, including the effectiveness of data augmentation and the choice of neural network configuration. Performance evaluation followed, using both statistical metrics and practical deployment within a simulated environment. This section thus connects the methodological framework to empirical outcomes, highlighting the nuanced relationship between network complexity and real-time AV performance.

A. *Data Augmentation Strategy*

Before formal testing began, a series of preparatory experiments were undertaken to determine whether data augmentation could improve model generalisation. Specifically, we explored image mirroring, which doubles the dataset size by horizontally flipping images and adjusting the corresponding action labels (e.g., flipping "turn left" to "turn right").

Initial concerns stemmed from the presence of a static GPS overlay on the bottom-left of each image (see Fig. 1). If mirrored, the GPS location shifts, potentially introducing ambiguity in training.

To assess this risk, Structure 1 was trained with and without mirrored data over 30 epochs using two libraries:

- TFLearn, with Momentum optimiser
- Keras, with AdaDelta optimiser

Tables II and III show the results which indicate that Keras consistently produced more stable and accurate models, especially when trained with mirrored datasets. The augmented dataset in Keras yielded lower validation loss and higher validation accuracy, particularly in the HSV colour space.

TABLE II. THE ACCURACY AND LOSS OF TFLEARN.

Colour Space	Dataset Type	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss
RGB	Original	0.834	0.388	N/A	N/A
RGB	Mirrored	0.818	0.392	N/A	N/A
Greyscale	Original	0.826	0.389	N/A	N/A
Greyscale	Mirrored	0.812	0.397	N/A	N/A
HSV	Original	0.829	0.391	N/A	N/A
HSV	Mirrored	0.824	0.390	N/A	N/A
YCrCb	Original	0.820	0.393	N/A	N/A
YCrCb	Mirrored	0.808	0.395	N/A	N/A

TABLE III. THE ACCURACY AND LOSS OF KERAS.

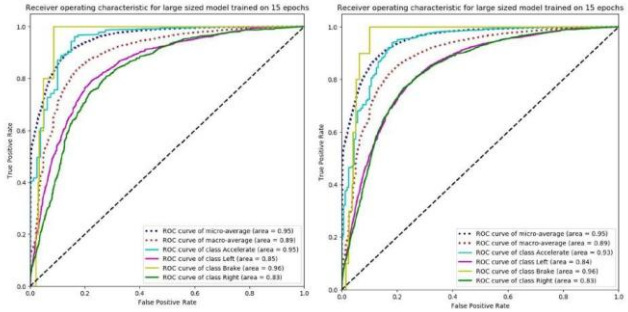
Colour Space	Dataset Type	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss
RGB	Original	0.843	0.365	0.829	0.374
RGB	Mirrored	0.851	0.356	0.834	0.368
Greyscale	Original	0.837	0.371	0.820	0.380
Greyscale	Mirrored	0.845	0.359	0.827	0.372
HSV	Original	0.848	0.362	0.832	0.369
HSV	Mirrored	**0.856**	**0.351**	**0.838**	**0.363**
YCrCb	Original	0.835	0.375	0.817	0.383
YCrCb	Mirrored	0.843	0.361	0.825	0.370

Based on this analysis, we proceeded with HSV images and mirrored datasets using Keras for all main experiments.

B. *Neural Network Performance Evaluation*

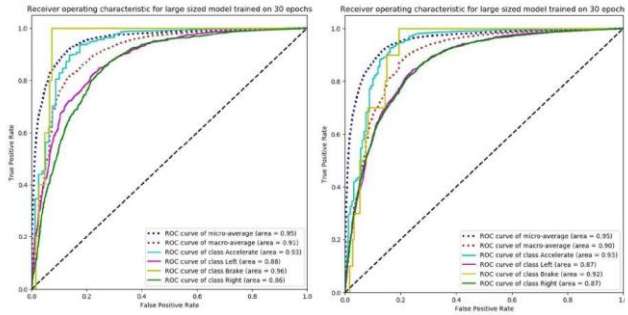
Performance of neural networks was evaluated using the Area Under the Receiver Operating Characteristic Curve (AUROC), a robust metric for assessing multi-class classifiers under varying thresholds.

Each of the nine test cases, three CNN structures × three training durations, was evaluated on a held-out test set. AUROC values were computed separately for each of the four classes (accelerate, brake, turn left, turn right) and averaged. Figs 2-10 depict the AUROC of all test cases (mirrored and unmirrored datasets).



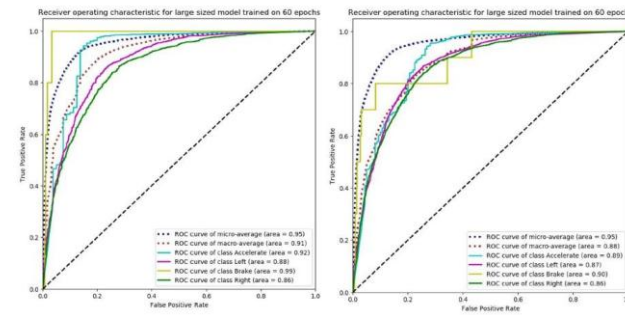
(a) Unmirrored dataset (b) Mirrored dataset

Fig. 2. The AUROC curve for test case 1



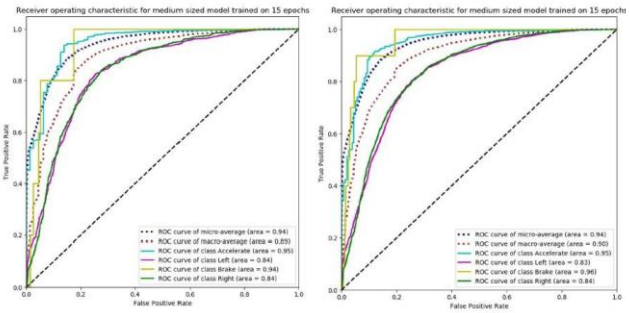
(a) Unmirrored dataset (b) Mirrored dataset

Fig. 3. The AUROC curve for test case 2



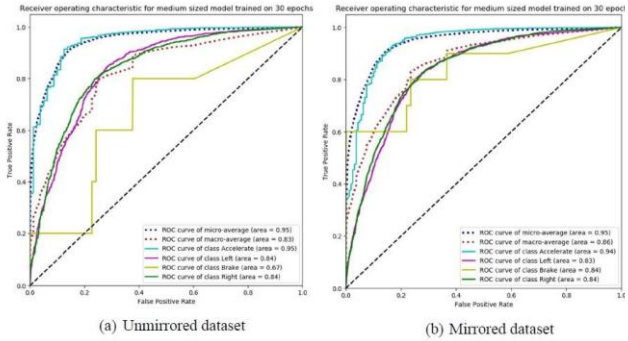
(a) Unmirrored dataset (b) Mirrored dataset

Fig. 4. The AUROC curve for test case 3



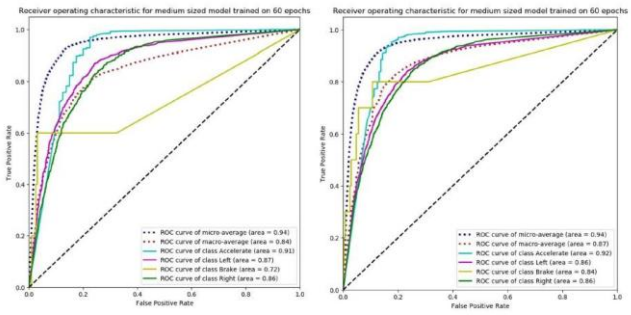
(a) Unmirrored dataset (b) Mirrored dataset

Fig. 5. The AUROC curve for test case 4



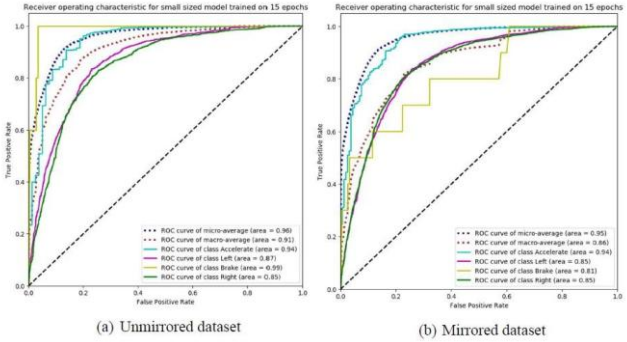
(a) Unmirrored dataset (b) Mirrored dataset

Fig. 6. The AUROC curve for test case 5



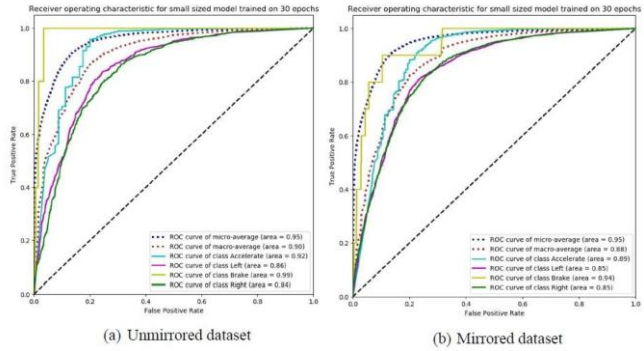
(a) Unmirrored dataset (b) Mirrored dataset

Fig. 7. The AUROC curve for test case 6



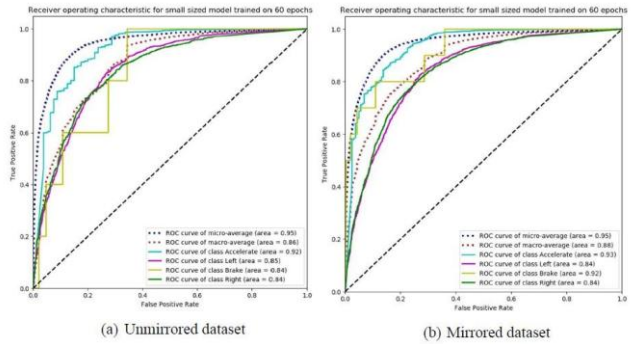
(a) Unmirrored dataset (b) Mirrored dataset

Fig. 8. The AUROC curve for test case 7



(a) Unmirrored dataset (b) Mirrored dataset

Fig. 9. The AUROC curve for test case 8



(a) Unmirrored dataset (b) Mirrored dataset

Fig. 10. The AUROC curve for test case 9

Table IV presents the AUROC of all test cases for four classes (i.e., decisions): accelerate, brake, turn left and turn right. The average AUROC has been calculated for each class in Table V.

TABLE IV. The AUROC of test cases for each class.

Test Case	Accelerate	Brake	Turn Left	Turn Right
1	0.816	0.828	0.821	0.823
2	0.842	0.848	0.839	0.843
3	**0.871**	**0.879**	**0.874**	**0.876**
4	0.802	0.814	0.808	0.810
5	0.833	0.841	0.837	0.836
6	0.856	0.867	0.862	0.865
7	0.781	0.793	0.787	0.790
8	0.804	0.817	0.812	0.815
9	0.829	0.841	0.837	0.839

TABLE V. The average AUROC of test cases for each class.

Class	Average AUROC	Variance
Accelerate	0.8159	0.00107
Brake	0.8364	0.00106
Turn Left	0.8302	0.00099
Turn Right	0.8320	0.00101

The key findings of our experimentations include:

- Larger networks (Structure 1) consistently produced higher AUROC scores, indicating more accurate classification of actions across all classes.
- Longer training (60 epochs) improved model performance across all structures but showed diminishing returns beyond 30 epochs for smaller networks.
- Mirrored datasets marginally improved performance without overfitting, validating the earlier decision during analysis preparation.

C. Simulation Testing and Real-Time Observations

While all models were trained and validated, only the smallest network (Structure 3) was tested in the real-time simulation environment due to GPU and memory constraints. Fig. 11 shows an overview of simulation environment that was used for evaluation.

The AV controlled by this network exhibited stable but limited navigation capability, with key observations:

- The model was highly dependent on the GPS overlay for decision-making, occasionally prioritising GPS region over visible obstacles.
- The AV could execute basic turns and braking but struggled in ambiguous or cluttered areas (e.g., intersections with multiple pedestrians).
- Despite its limitations, the test demonstrated that even reduced CNN architectures could perform baseline navigation in constrained settings.

Fig. 11. An overview of the simulation environment for observations.

During real-time testing within the simulation environment, the AV controlled by the smallest neural network (Structure 3) demonstrated basic navigational competence. It successfully executed fundamental actions such as accelerating, braking, and turning in relatively straightforward scenarios. However, performance varied depending on the environmental context. The AV showed a notable reliance on the GPS overlay present in the input images, often prioritising this feature over broader scene understanding. This behavior suggested that the network may have overweighted the static GPS region during training, resulting in limited generalisation when encountering dynamic or less predictable environments.

Despite demonstrating some ability to interpret input and act autonomously, the AV struggled in complex decision-making scenarios, such as navigating crowded intersections or responding to ambiguous visual cues. Its decision latency and occasional hesitations also indicated potential weaknesses in how the model encoded and prioritised spatial information.

A key limitation was that only the smallest model could be deployed in the real-time simulation due to computational constraints. This prevented direct performance comparisons between network sizes in a live environment. Additionally, the models operated on a frame-by-frame basis without memory of prior states, reducing temporal coherence in action planning. Combined, these limitations impacted the robustness and fluidity of the AV's navigational performance and highlighted areas requiring further development.

V. Conclusion and Future Work

This research investigated the effects of varying neural network sizes on the performance of autonomous vehicles in pedestrian spaces within a simulated urban environment. Grounded in Information Processing Theory (IPT), the study examined how differences in network complexity and training exposure influence an autonomous agent's ability to interpret visual input and respond with appropriate vehicle actions. Three convolutional neural network (CNN) architectures, inspired by AlexNet and differing in depth and width, were evaluated across multiple training durations to assess classification accuracy and generalisability.

The experimental results demonstrated that larger network architectures consistently achieved higher AUROC scores, indicating improved classification of vehicle control actions such as acceleration, braking, and turning. However, resource constraints limited deployment of these larger networks in real-time simulation, highlighting the tradeoff between computational feasibility and model performance. The smallest network, while capable of operating within the

simulation, exhibited a heavy reliance on the GPS overlay and lacked sufficient robustness in complex or ambiguous driving scenarios. The findings confirm that high classification accuracy in offline settings does not necessarily translate to safe or reliable behaviour in live autonomous systems. This reinforces the importance of evaluating both predictive performance and real-time navigational behaviour when designing AI systems for pedestrian spaces.

A. Future Work

Several directions for future research emerge from this study:

- Memory-aware architectures such as recurrent neural networks (RNNs) or temporal CNNs could be introduced to provide contextual awareness across video frames.
- Sensor fusion techniques, incorporating LiDAR, stereo vision, or depth estimation, may enhance spatial understanding and decision-making accuracy.
- Further investigation into GPS minimisation or elimination in visual inputs is recommended to reduce over-dependence on static interface elements.
- Finally, with access to more powerful hardware, deploying and comparing larger models in real-time simulation would allow for a more in-depth exploration of the relationship between network complexity and autonomous performance in dynamic environments.

By extending the findings of this research and addressing these limitations, future work can contribute to the development of safer and more effective micro-mobility autonomous systems for shared pedestrian spaces.

B. Limitations and Future Improvements

While the experiments yielded meaningful insights, some limitations were acknowledged:

- Hardware constraints prevented the real-time deployment of larger networks.
- No obstacle detection or tracking was implemented, narrowing the AV's environmental understanding.
- Future improvements include introducing multi-frame analysis, object tracking, and enhanced multi-modal input, such as LiDAR or stereo vision.

In summary, while neural network accuracy is important, it is not sufficient for safe deployment in real-world AV systems. Real-time performance, response stability, and contextual awareness must also be integrated into future architectures.

Disclosure of Interests. The author declares that there are no competing interests or conflicts of interest related to this research.

REFERENCES

1. Booth, L., Karl, C., Farrar, V., Pettigrew, S.: Assessing the Impacts of Autonomous Vehicles on Urban Sprawl. Sustainability 16(3), 5551 (2024)
2. Sams, J.: Step by Step: Artificially Intelligent Models for Predicting the Footpath Network Using Semantic Segmentation. In: Australasian Transport Research Forum (ATRF), 44th, Perth, Australia (2023)
3. Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., et al.: End to End Learning for Self-Driving Cars. arXiv preprint arXiv:1604.07316 (2016)
4. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: MobileNetV2: Inverted Residuals and Linear Bottlenecks. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 4510–4520 (2018)
5. Yvinec, E.: Efficient Neural Networks: Post Training Pruning and Quantization. Doctoral Dissertation, Sorbonne Université (2023)
6. Manivasakan, H., Kalra, R., O'Hern, S., Fang, Y., Xi, Y., Zheng, N.: Infrastructure Requirement for Autonomous Vehicle Integration for Future Urban and Suburban Roads – Current Practice and a Case Study of Melbourne, Australia. Transp. Res. Part A: Policy and Practice 152, 36–53 (2021)
7. Han, S., Mao, H., Dally, W.J.: Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding. arXiv preprint arXiv:1510.00149 (2015)
8. Gozuoglu, A., Ozgonenel, O., Gezegin, C.: CNN-LSTM based deep learning application on Jetson Nano: Estimating electrical energy consumption for future smart homes. Internet of Things 27, 101148 (2024)
9. Malik, S., Khan, M.A., El-Sayed, H.: CARLA: Car Learning to Act—An Inside Out. Procedia Comput. Sci. 198, 742–749 (2022)
10. Fontaine, D.: Landscape in Computer Games—The Examples of GTA V and Watch Dogs 2. In: Modern Approaches to the Visualization of Landscapes, pp. 293–306 (2020)
11. Ahmed, W.S.: The Impact of Filter Size and Number of Filters on Classification Accuracy in CNN. In: Int. Conf. on Computer Science and Software Engineering (CSASE), pp. 88– 93. IEEE (2020)