Deep Learning Architecture to Assist with Steering a Powered Wheelchair

Malik J. Haddad, *Member, IEEE* and David A. Sanders *MBE TD VR*

***Abstract*—This paper describes a novel Deep Learning architecture to assist with steering a powered wheelchair. A rule-based approach is utilized to train and test a Long Short Term Memory (LSTM) Neural Network. It is the first time a LSTM has been used for steering a powered wheelchair. A disabled driver uses a joystick to provide desired speed and direction, and the Neural Network provides a safe direction for the wheelchair. Results from the Neural Network are mixed with desired speed and direction to avoid obstacles. Inputs originate from a joystick and from three ultrasonic transducers attached to the chair. The resultant course is a blend of desired directions and directions that steer the chair to avoid collision. A rule-based approach is used to create a training and test set for the Neural Network system and applies deep learning to predict a safe route for a wheelchair. The user can over-ride the new system if necessary.**

***Index Terms*—Deep Learning, Neural Network, Rule-based, Disabled, Steer, Wheelchair.**

I. Introduction

THIS paper presents a novel architecture to provide a safe steering direction for a powered wheelchair. The architecture applies a rule-based approach to generate an input set to train and test a Long Short Term Memory (LSTM) Neural Network. The research is part of broader research conducted by the authors based on [1]. The novel architecture is compared to the system presented in [2] and is applied to three scenarios. Two of these scenarios were considered in [2]. Results are compared and advantages of the new architecture are presented.

The World Health Organization's (WHO) report on disability indicated that around one sixth of the world population were suffering of some sort of disability and 2 to 4% of them were subject to significant difficulties in mobility. These numbers were greater than previous WHO estimates due to population ageing, fast spread of chronic disorders, and advances in modern medical treatment [3]. In many cases, people with disabilities struggled with daily manoeuvring tasks and often relied on helpers and carers for other daily activities [4].

During the past three decades, many researchers created systems that aimed to help disabled powered wheelchair users to navigate their wheelchairs safely. Song and Chen [5] applied asymmetric mapping and ultrasonic sensors for wheelchair navigation. Lee [6] used infrared light reflection for wheelchair localization. Sanders *et al.* [7] created a system that improved driving with sensors that controlled veer. Langner [8] created a scanning collision avoidance device (SCAD) based on a single rotating ultrasonic transducer. Sanders and Bausch [9] created an expert system that improved steering of a powered wheelchair by interpreting users' hand tremor. Most of the research aimed at helping physically disabled powered wheelchair drivers to maneuver their wheelchairs safely and enhance their quality of life. The research often assumed powered wheelchair users were cognitively aware of their surroundings but lacked the physical ability to drive their wheelchairs. Over the years, the type of disability of powered wheelchair users shifted from mostly physically disabled to mostly mentally disabled or mentally and physically disabled. The older systems successfully helped physically disabled users to drive their wheelchairs but did not provide much help for mentally or physically and mentally disabled users.

To help mentally or physically and mentally disabled users, researchers have used more advanced methods and approaches for example Artificial Intelligence (AI) and Multiple Criteria Decision Making (MCDM). Sanders [10] created a system based on self-reliance factors to share control between human powered wheelchair users and ultrasonic sensor system. Sanders *et al.* [11] created a rule-based system to choose a steering direction of a powered wheelchair. Haddad *et al*. [12] created a system based on ultrasonic sensor readings and combined MCDM with vector algebra to provide a safe steering direction. Haddad and Sanders [2] used PROMETHEE II, a MCDM method to recommend a best-compromise path. Haddad *et al.* [13] created an intelligent Human Machine interface (HMI) and control for steering a powered wheelchair using a Raspberry Pi microcomputer.

More advanced and sophisticated AI algorithms could be used and applied to improve the quality of life of mentally or physically and mentally disabled powered wheelchair users by helping them drive their powered wheelchairs safely. That could increase their self-reliance and self-confidence, and reduce the need for help from carers and helpers. A system is presented here that uses a rule-based approach to generate an input set to train and test a LSTM Neural Network is described. The LSTM Neural Networks used deep learning to classify

Line for date paper submitted for review.

M. J. Haddad is a Senior Research Fellow and D. A. Sanders is a Professor at the University of Portsmouth, UK (e-mail:malik.haddad@port.ac.uk, david.sanders@port.ac.uk).

This work was supported by an EPSRC EP/S005927/1 project titled "Using artificial intelligence to share control of a powered-wheelchair between a wheelchair user and an intelligent sensor system". Investigators: D. Sanders, and A. Gegov. Senior Researcher: M. J. Haddad.

input data (data from three ultrasonic sensors) to six steering directions. Sensor readings were sent through different layers of the network, with each layer defining features of the sensor readings. After data were processed through the layers, the system identified the appropriate identifiers for classifying the sensor reading to six classes that represented different steering directions for a powered wheelchair: Right spin, Right turn, Left spin, Left turn, Forward, and Stop.

The rule-based approach is briefly explained in Section II. Section III briefly describes LSTM Neural Networks, presents the LSTM Network architecture used in this research, the training and testing of the LSTM Network used, and then considers three real world scenarios to test the Network. Section IV presents the approach used to mix the Neural Network outputs with the joystick inputs. Results are described within Section V and some Conclusions and Future Work are described within Section VI.

II. The Rule-Based Approach

Researchers have considered different types of sensors to assist wheelchairs with driving safely. Sanders *et al.* [14] presented sensors to help wheelchair drivers drive their wheelchairs safely, including infrared [15]; ultrasonic [16], and structured light [17]. Global systems demonstrated poor performance inside buildings [18]. Other local architectures have been successfully applied, including: odometers, tilt sensors, gyroscopes, or ultrasonics [19], [20] & [21]. This paper used ultrasonic sensors similar to those considered in [2]. Three sensors were used and were mounted on to the front of the wheelchair. The first sensor measured distance from the nearest obstacle on the right of the chair, the second sensor measured distance from the nearest obstacle to the front of the wheelchair, and the third sensor measured the distance from the nearest obstacle on the left.

The ultrasonic sensors were studied and tested using different objects to create polar plots. Due to their physical structure, these sensors did not suffer from cross talk and side lobe interference.

An example of the polar plot of the ultrasonic sensors used is shown in Fig. 1. Distance was estimated by measuring time for a pulse to travel towards an obstacle and back again [22].

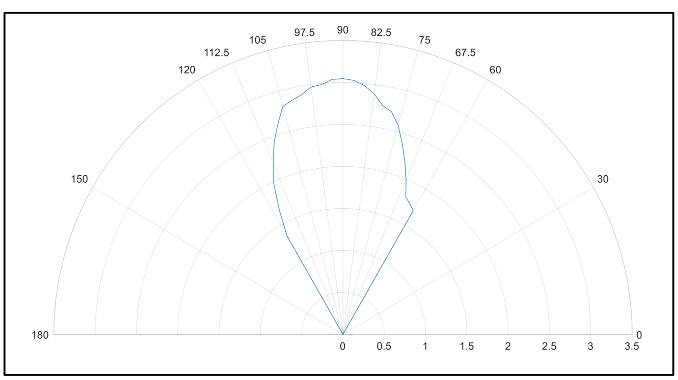


Fig. 1. Polar plot for HC SR04 ultrasonic sensor detecting an object.

A matrix was over-laid on the area in front of each sensor. Each matrix consisted of four elements: Adjacent, Nearby, Distant, and Faraway as shown in Figs 2, 3 & 4. Sensor readings identified the distance between the wheelchair and the nearest obstacle. The readings were: Distance from an obstacle to the right of the wheelchair (D_r), Distance to the center (D_c), and Distance to the left (D_l). When no object was in the sensor range the distance was set to Faraway.

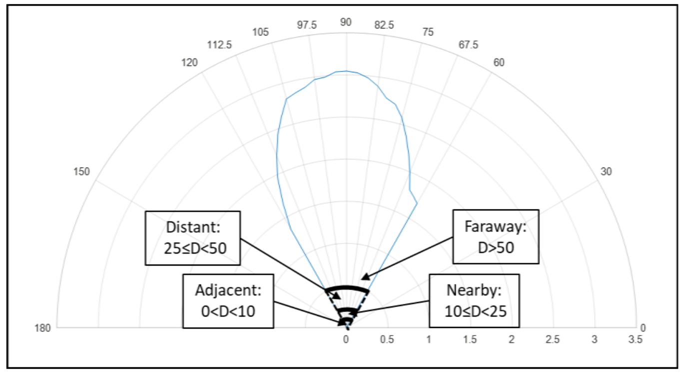


Fig. 2. Four-element matrix layout of the area in front of an ultrasonic sensor.

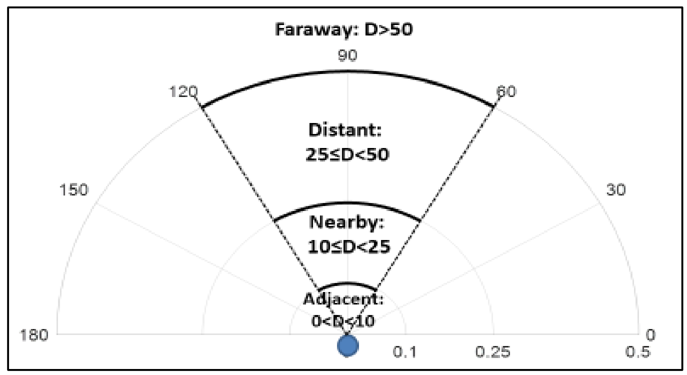


Fig. 3. Close up view of the four-element matrix layout of the area in front of an ultrasonic sensor shown in Fig. 2.

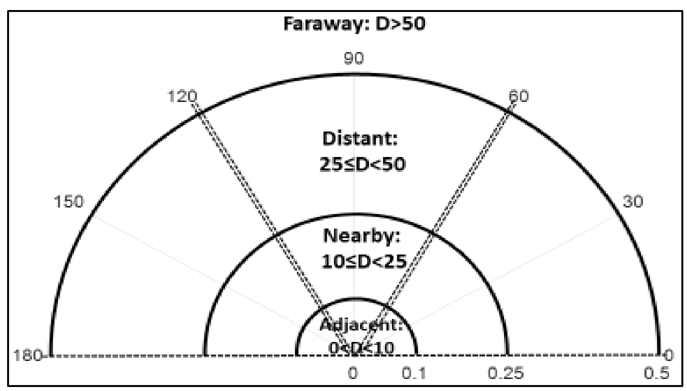


Fig. 4. Configuration of the array of three ultrasonic sensors used in this paper.

Based on the configuration shown in Fig. 4 a rule-based approach was used to deduce an overall direction for the powered wheelchair depending on the sensor readings as shown in Table I.

TABLE I
DEDUCED RULES FOR AN OVERALL DIRECTION FOR A POWERED WHEELCHAIR

D_r	D_c	D_l	Overall Direction
D_r>50	D_c>50	D_l>50	Forward
D_r>50	D_c>50	25<D_l<50	Forward
D_r>50	D_c>50	10<D_l<25	Right turn

$D_r>50$	$D_c>50$	$D_l<10$	Right spin
$D_r>50$	$25<D_c<50$	$D_l>50$	Right turn
$D_r>50$	$25<D_c<50$	$25<D_l<50$	Right turn
$D_r>50$	$25<D_c<50$	$10<D_l<25$	Right spin
$D_r>50$	$25<D_c<50$	$D_l<10$	Right spin
$D_r>50$	$10<D_c<25$	$D_l>50$	Right turn
$D_r>50$	$10<D_c<25$	$25<D_l<50$	Right turn
$D_r>50$	$10<D_c<25$	$10<D_l<25$	Right spin
$D_r>50$	$10<D_c<25$	$D_l<10$	Right spin
$D_r>50$	$D_c<10$	$D_l>50$	Right spin
$D_r>50$	$D_c<10$	$25<D_l<50$	Right spin
$D_r>50$	$D_c<10$	$10<D_l<25$	Stop
$D_r>50$	$D_c<10$	$D_l<10$	Stop
$25<D_r<50$	$D_c>50$	$D_l>50$	Forward
$25<D_r<50$	$D_c>50$	$25<D_l<50$	Forward
$25<D_r<50$	$D_c>50$	$10<D_l<25$	Right turn
$25<D_r<50$	$D_c>50$	$D_l<10$	Right turn
$25<D_r<50$	$25<D_c<50$	$D_l>50$	Left turn
$25<D_r<50$	$25<D_c<50$	$25<D_l<50$	Forward
$25<D_r<50$	$25<D_c<50$	$10<D_l<25$	Right turn
$25<D_r<50$	$25<D_c<50$	$D_l<10$	Right spin
$25<D_r<50$	$10<D_c<25$	$D_l>50$	Left turn
$25<D_r<50$	$10<D_c<25$	$25<D_l<50$	Left turn
$25<D_r<50$	$10<D_c<25$	$10<D_l<25$	Right turn
$25<D_r<50$	$10<D_c<25$	$D_l<10$	Right spin
$25<D_r<50$	$D_c<10$	$D_l>50$	Left spin
$25<D_r<50$	$D_c<10$	$25<D_l<50$	Left spin
$25<D_r<50$	$D_c<10$	$10<D_l<25$	Stop
$25<D_r<50$	$D_c<10$	$D_l<10$	Stop
$10<D_r<25$	$D_c>50$	$D_l>50$	Right turn
$10<D_r<25$	$D_c>50$	$25<D_l<50$	Right turn
$10<D_r<25$	$D_c>50$	$10<D_l<25$	Forward
$10<D_r<25$	$D_c>50$	$D_l<10$	Forward
$10<D_r<25$	$25<D_c<50$	$D_l>50$	Left spin
$10<D_r<25$	$25<D_c<50$	$25<D_l<50$	Left turn
$10<D_r<25$	$25<D_c<50$	$10<D_l<25$	Forward
$10<D_r<25$	$25<D_c<50$	$D_l<10$	Right turn
$10<D_r<25$	$10<D_c<25$	$D_l>50$	Left spin
$10<D_r<25$	$10<D_c<25$	$25<D_l<50$	Left turn
$10<D_r<25$	$10<D_c<25$	$10<D_l<25$	Stop
$10<D_r<25$	$10<D_c<25$	$D_l<10$	Stop
$10<D_r<25$	$D_c<10$	$D_l>50$	Left spin
$10<D_r<25$	$D_c<10$	$25<D_l<50$	Left spin
$10<D_r<25$	$D_c<10$	$10<D_l<25$	Stop
$10<D_r<25$	$D_c<10$	$D_l<10$	Stop
$D_r<10$	$D_c>50$	$D_l>50$	Left spin
$D_r<10$	$D_c>50$	$25<D_l<50$	Left turn
$D_r<10$	$D_c>50$	$10<D_l<25$	Forward
$D_r<10$	$D_c>50$	$D_l<10$	Forward
$D_r<10$	$25<D_c<50$	$D_l>50$	Left spin
$D_r<10$	$25<D_c<50$	$25<D_l<50$	Left spin
$D_r<10$	$25<D_c<50$	$10<D_l<25$	Right turn
$D_r<10$	$25<D_c<50$	$D_l<10$	Forward
$D_r<10$	$10<D_c<25$	$D_l>50$	Left spin
$D_r<10$	$10<D_c<25$	$25<D_l<50$	Left spin
$D_r<10$	$10<D_c<25$	$10<D_l<25$	Stop
$D_r<10$	$10<D_c<25$	$D_l<10$	Forward
$D_r<10$	$D_c<10$	$D_l>50$	Stop
$D_r<10$	$D_c<10$	$25<D_l<50$	Stop
$D_r<10$	$D_c<10$	$10<D_l<25$	Stop
$D_r<10$	$D_c<10$	$D_l<10$	Stop

M.S. Excel was used to produce five thousand randomly generated values for D_r, D_c, and D_l. Table I was used to give each set of D_r, D_c, and D_l an Overall Direction to create a (5000x4) matrix.

III. Long Short Term Memory (LSTM) Neural Network

LSTM Neural Networks are often considered as a branch of Recurrent Neural Network (RNN). They were introduced by Hochreiter and Schmidhuber [23], since then, researchers have worked on simplifying their architecture, and improving their efficiency and accuracy [24] & [25]. LSTM have been successfully applied to handwriting recognition problems, text completion, and many other problems [26].

A. LSTM Architecture Used

The LSTM architecture used in this paper is shown in Fig. 5. It considered five layers:

1) Sequence Input Layer with three inputs.
2) Bilstm Layer(100,'OutputMode','sequence')
3) Fully Connected Layer with six nodes.
4) Softmax Layer.
5) Classification Layer.

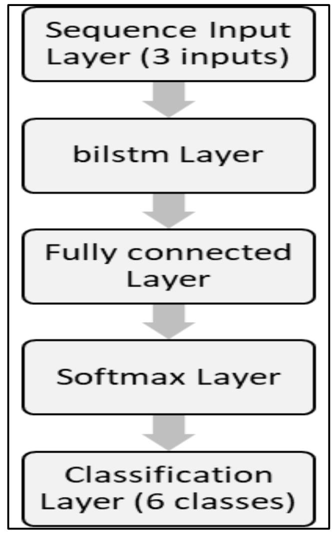


Fig. 5. LSTM Neural Network used in this paper; 3 inputs, 6 output classes.

Fig. 6 shows a screen shot of the MATLAB code used to create the layers of the LSTM Network used in this research and to set the training options of the Network.

```
>> layers = [ ...
    sequenceInputLayer(3)
    bilstmLayer(100,'OutputMode','sequence')
    fullyConnectedLayer(6)
    softmaxLayer
    classificationLayer];
>> options = trainingOptions('adam', ...
    'Plots','training-progress', ...
    'InitialLearnRate',0.01, ...
    'MaxEpochs',100);
```

Fig. 6. MATLAB code used to create layers of the LSTM Neural Network and to set the training options of the Network used in this paper.

Using the MATLAB code shown in Fig. 6 a LSTM Neural Network was created with 3 inputs, 100 hidden units in the bilstm Layer, and 6 output classes. An adaptive momentum estimation algorithm was used in this architecture with an initial learning rate of 0.01 and maximum number of epochs of 100.

B. Training and Testing of the Network

Training and testing of the LSTM Network was conducted on a MATLAB platform. The (5000x4) matrix was imported to MATLAB and used as training and testing sets. Fig. 7 shows a screen shot of the MATLAB code used to split the (5000x4) matrix in a ratio 7:3 for training and testing sets respectively.

```
>> Dr5=set5.Dr;
>> Dc5=set5.Dc;
>> Dl5=set5.Dl;
>> X5=[Dr5 Dc5 Dl5]';
>> Y5=set5.Direction;
>> Y5=Y5';
>> XTrain5=X5(:,1:3500);
>> XTest5=X5(:,3501:5000);
>> YTrain5=Y5(:,1:3500);
>> YTest=Y5(:,3501:5000);
```

Fig. 7. Screen shot of the MATLAB code used to create separate training and testing sets from (5000x4) matrix.

Fig. 8 shows Network training progress with an initial learning rate 0.01 and 100 epochs, as Network training progressed. The Network accuracy increased to around 97%.

Fig. 9 shows the training result. Fig. 10 shows that Network accuracy reached 96.87% when tested with the testing set (shown in a red oval in Fig. 10). Fig. 11 shows the confusion chart produced from testing the Network with the testing set.

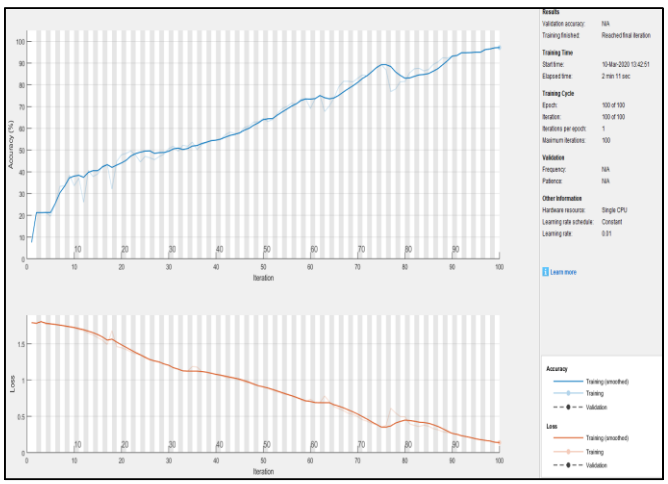


Fig. 8. Screen shot of Network training progress Network accuracy increasing and Network loss decreasing.

```
>> net5 = trainNetwork(XTrain5,YTrain5,layers,options);
Training on single CPU.
|======================================================================================|
|  Epoch  |  Iteration  |  Time Elapsed  |  Mini-batch  |  Mini-batch  |  Base Learning  |
|         |             |   (hh:mm:ss)   |   Accuracy   |     Loss     |      Rate       |
|======================================================================================|
|       1 |           1 |       00:00:06 |        7.54% |       1.7919 |          0.0100 |
|      50 |          50 |       00:01:08 |       64.51% |       0.9061 |          0.0100 |
|     100 |         100 |       00:02:11 |       97.14% |       0.1370 |          0.0100 |
|======================================================================================|
```

Fig. 9. Screen shot of Network training outcome.

```
>> nnz(testPred == YTest)/numel(YTest)

ans =

    0.9687
```

Fig. 10. Screen shot of Network accuracy when tested with the testing set.

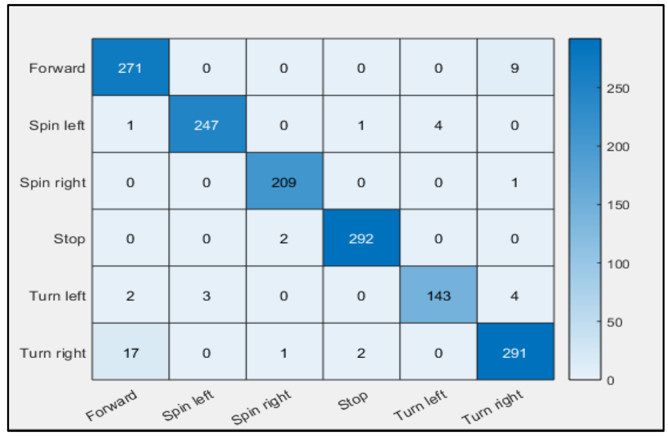


Fig. 11. Screen shot of the confusion chart used to assess Network accuracy.

Fig. 9 shows that the Network required 2 minutes and 11 seconds to complete 100 epochs with an initial learning rate of 0.01. The Network produced high accuracy as shown in Fig. 10. Different values for initial learning rate and maximum number of epochs were considered. A best compromise between learning time needed and overall accuracy of the Network was conducted. The initial learning rate and maximum number of epochs was set to 0.01 and 100 respectively.

C. Real World Testing of the System

The trained and tested Network was used to provide an overall outcome based on ultrasonic sensor readings. Three scenarios were considered as a powered wheelchair moved through a setting with some boxes as obstacles as shown in Fig. 12. Scenarios 1 & 2 were similar to those considered in [2].

1) Scenario 1: No object detected (Location A in Fig. 12).
2) Scenario 2: Object detected to the right (Location B in Fig 12).
3) Scenario 3: Objects detected to the left and also ahead (Location C in Fig 12).

Six options for the overall direction of a powered wheelchair were considered: Left turn, Left spin, Forward, Right turn, Right spin, and Stop.

1) Scenario 1: (Location A in Fig. 12)

As the chair started moving, nothing was within range of the sensors. All the distances were set to Faraway.

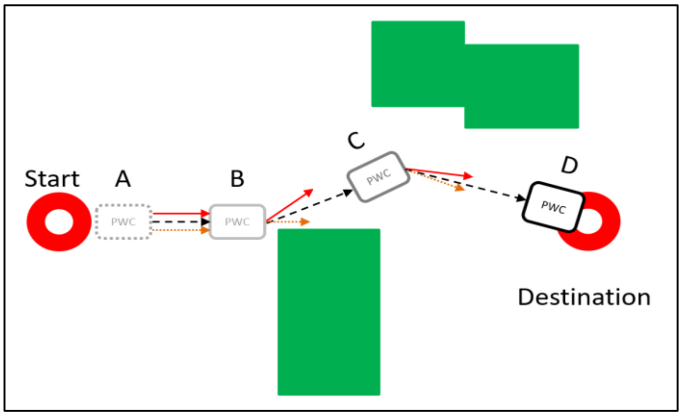
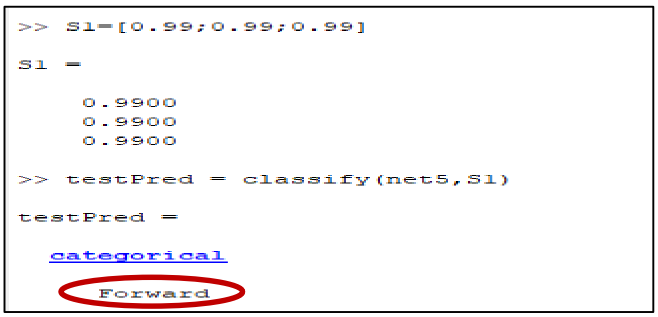


Fig. 12. Powered wheelchair driving through a setting containing obstacles.

A screen shot of the MATLAB code used to apply the Network for scenario 1 is shown in Fig. 13. The Network output was “Forward” and highlighted by a red oval in Fig. 13.

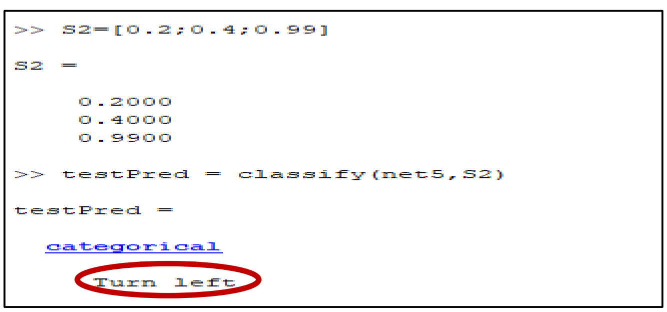


```
>> S1=[0.99;0.99;0.99]

S1 =

    0.9900
    0.9900
    0.9900

>> testPred = classify(net5,S1)

testPred =

  categorical

     Forward
```

Fig. 13. Screen shot of MATLAB code for scenario 1. The Network output was Forward.

2) Scenario 2: (Location B in Fig. 12)

An object was detected to the right as the wheelchair moved forward. A screen shot of the MATLAB code used to apply the Network for scenario 2 is shown in Fig. 14. The Network output was “Left turn”, highlighted by a red oval in Fig. 14.

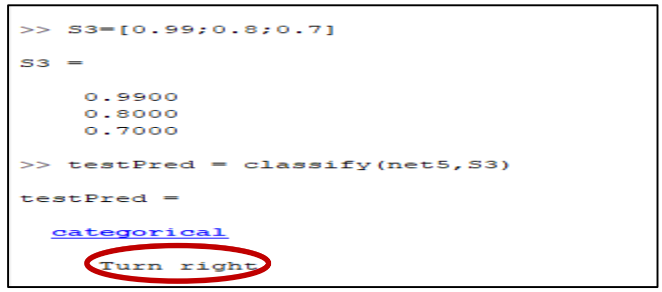


```
>> S2=[0.2;0.4;0.99]

S2 =

    0.2000
    0.4000
    0.9900

>> testPred = classify(net5,S2)

testPred =

  categorical

     Turn left
```

Fig. 14. Screen shot of MATLAB code for scenario 2. The Network output was Left turn.

3) Scenario 3: (Location C in Fig. 12)

The wheelchair moved away from the obstacle and the chair moved in a new direction. Two more obstacles were detected at a far distance to the left and in front of the wheelchair as shown in position C in Fig. 12. A screen shot of the MATLAB code used to apply the Network for scenario 3 is shown in Fig. 15. The Network output was “Right turn” and is highlighted by a red oval in Fig. 15.



```
>> S3=[0.99;0.8;0.7]

S3 =

    0.9900
    0.8000
    0.7000

>> testPred = classify(net5,S3)

testPred =

  categorical

     Turn right
```

Fig. 15. Screen shot of MATLAB code for scenario 3. The Network output was Right turn.

IV. Mixing Network Output with Joystick Input

This research combined human driving skill and autonomy with intervention from the ultrasonic sensors if they were required. A joystick provided an interface between the powered wheelchair users and their wheelchairs for control of speed and direction. Disabled drivers could use their skill to safely drive, but the sensors were often more accurate, and they could balance any lack of ability or awareness. When in varying or complicated environments then the ultrasonic sensors often provided better choices about courses to follow. In all three scenarios considered in this paper, the user joystick was held in a position aiming to reach the destination point shown in Fig. 12 by taking the shortest path without considering obstacles in the surrounding environment.

V. Results

The LSTM Neural Network presented in this paper showed successful outcomes when tested. Sensors measured the distance from the nearest obstacle to the right, center, and left of a wheelchair. Sensor readings were used as inputs for the LSTM Network. Fig. 16 shows the resultant direction when mixing the LSTM Network output for scenario 1 with nothing being detected and the joystick pushed forward. The solid red arrow is the output from the LSTM Network, the dotted orange arrow is the joystick input, and the dashed black arrow is the resultant bearing and speed.

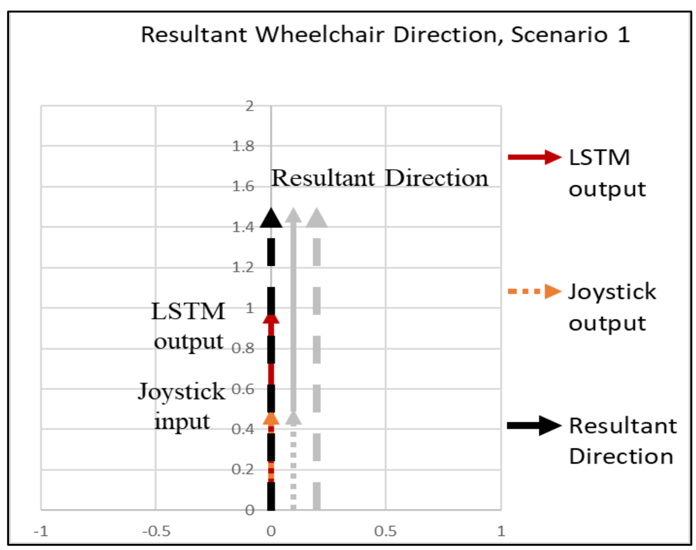


Fig. 16. Resultant direction of the Wheelchair for scenario 1 after mixing the LSTM Network output with joystick input.

Fig. 17 shows the resultant direction when mixing the LSTM Network output for scenario 2 with one obstacle detected on the right and the input form the joystick when pushed forward. The solid red arrow is the output from the LSTM Network, the dotted orange arrow is the input from the joystick, and the dashed black arrow is the resultant speed and bearing.

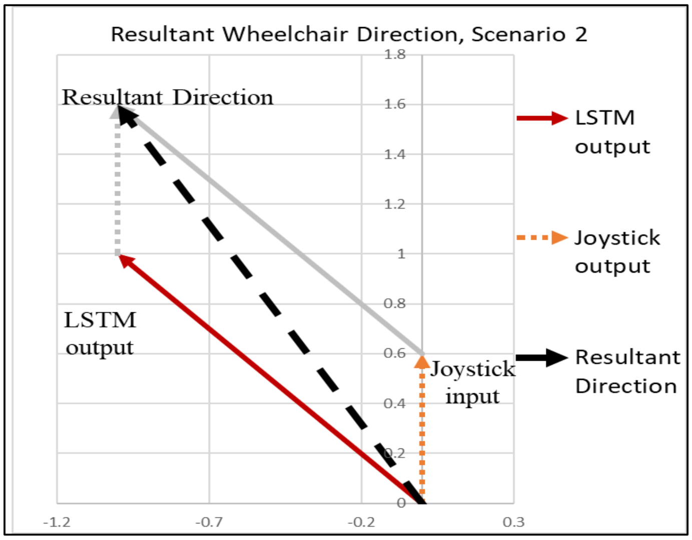


Fig. 17. Resultant direction of the Wheelchair for scenario 2 after mixing the LSTM Network output with joystick input.

Fig. 18 shows the resultant direction when mixing the LSTM Network output for scenario 3. The obstacle detected in scenario 2 was no longer detected. Instead, as the wheelchair moved in the new direction, two objects were sensed on the left and in front of the wheelchair. The joystick was pushed right toward the destination point. The solid red arrow is the output from the LSTM Network, the dotted orange arrow is the input from the joystick, and the dashed black arrow is the resultant direction and speed.

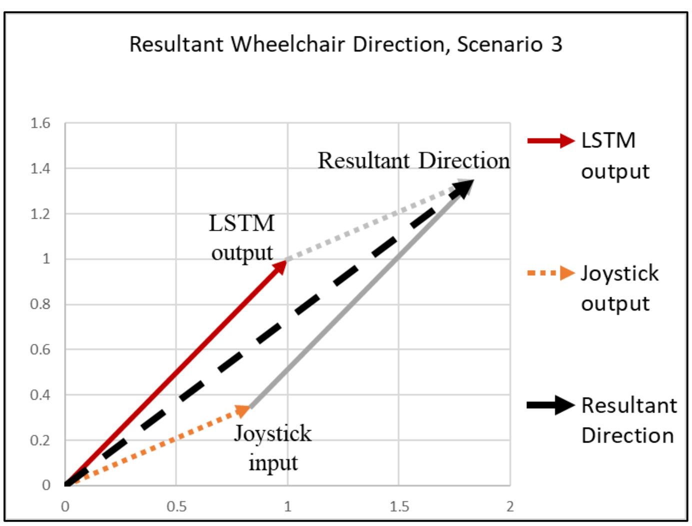


Fig. 18. Resultant direction of the Wheelchair for scenario 3 after mixing the LSTM Network output with joystick input.

VI. Discussion

Work presented in this paper could bring benefits to society by improving mobility. The new approach could be used to enhance the quality of life and improve mobility for children and young people with multiple-sclerosis, arthritis, stroke, paraplegia, orthopedic-impairment, cerebral-palsy and diabetes and will be useful for people with missing or damaged-limbs. The work will benefit disabled and older users and children in schools and institutions, making a significant positive difference, especially for people with limited spatial awareness or cognitive ability. The research could introduce some autonomy and reduce the need for carers.

The new architecture successfully mixed input from a joystick with sensors to maintain the autonomy of the driver. Wheelchair drivers successfully controlled the direction of their chairs while the sensor systems handled obstacle avoidance. The sensors provided a safe path as the wheelchair moved.

Results from scenarios 1 and 2 demonstrated that the system provided a safe direction and steered the chair away from obstacles. Results from scenario 3 demonstrated the fast sensor update time and quick system response to changes in the surrounding environment of the wheelchair.

If obstacles were further apart or there were fewer of them, then drivers did not require help. If a driver was provided with higher authority in those cases, then satisfaction and performance was improved. If there were many obstacles or nothing close to the chair, then the input from the joystick was successfully modified to avoid collision.

The environment surrounding a powered wheelchair can include many obstacles and obstacles can suddenly appear. The new approach used a rule based approach to provide inputs to the Neural Network. The rule based approach provided robust inputs to the ANN and the ANN produced dynamic and reliable outcomes and would provide quick responses when obstacles suddenly appeared.

Since the new approach was created using ANNs, Transfer Learning (TL) could be applied to transform the new approach [27]. The new approach could be generalized to fit any type of powered wheelchair.

VII. Conclusions and Future Work

Systems presented here provided a faster and more dynamic response to obstacles than the system presented in [2] and successfully steered chairs away from obstacles.

The new system could learn to steer a powered wheelchair in new environments as opposed to the system presented in [2]. The new system introduced some autonomy and potentially reduced the need for helpers by using a simple and computationally inexpensive LSTM Neural Network.

The output from the LSTM Network could be over-ridden if a joystick was held still in a position. Joystick input was integrated so that the system would eventually be over-ridden by the disabled user. If nothing was detected then a chair tended to drive as directed by a user through the joystick.

Results from testing the LSTM Network confirmed that it performed satisfactorily. The new approach will be extensively tested to ensure safety concerns have been answered and drivers' safety standards have been met before clinically trialed at Chailey Heritage foundation.

Many researchers successfully applied AI techniques for powered mobility such as probabilistic AI techniques under uncertainty [28], Learning from Demonstration [29], continuous path refinement using covariant gradient techniques [30] and probabilistic graphical model for natural language commands [31]. The work presented in this papers applied ANNs to powered wheelchair navigation and improved user self-confidence and reliance by increasing autonomy and providing a safe steering direction using dynamic and simple yet effective AI techniques.

A reason for the work was to reduce cost. The new approach provided safe and reliable results and increased user autonomy so that the need for and cost of carers will be reduced. Authors are considering uploading the program to an open access platform were users can download and use the new approach free of charge."

The research is now exploring the adjustment of pre-planned routes [32], force sensing [33], analyzing performance both with ultrasonic sensors connected, and when they are disconnected [34] and analyzing the effect of time delays on driving performance [35].

The authors will consider different input devices for example lever switches and head or chin switches. The authors will investigate applying Neural Networks for this application using different programming languages such as Python and R.

Future work will consider different types of AI applied to wheelchair problems. More advanced and computationally cheap AI techniques will be investigated. More sophisticated AI Algorithms will be considered and other types of Neural Networks could be used. The authors will consider overall directions. A deeper LSTM Network will be investigated with more layers and other types of sensors will be investigated for example, video cameras laser and infrared sensors.

REFERENCES

[1] Sanders, and A. Gegov, "Using artificial intelligence to share control of a powered-wheelchair between a wheelchair user and an intelligent sensor system", *EPSRC Project 2019 – 2022,* 2018.

[2] M. J. Haddad and D. A. Sanders, "Selecting a Best Compromise Direction for a Powered Wheelchair Using PROMETHEE", *IEEE T Neur. Sys. Reh.,* vol. 27, no. 2, pp. 228-235, Feb. 2019.

[3] M. K. Joshi, M. V. Gupta, M. M. Gosavi,, and M. S. Wagh, "A multifunctional smart wheelchair," *Int. J. Adv. Res. Electron. Commun. Eng.,* vol. 4, no. 5, pp. 1281-1284, 2015.

[4] J. Leaman, and H. M. La, "A comprehensive review of smart wheelchairs: past, present, and future*," IEEE T Hum. M-Mach. Syst.,* vol. 47, no. 4, pp. 486-499, 2017.

[5] K. T. Song, and C. C. Chen, "Application of asymmetric mapping for wheelchair navigation using ultrasonic sensors," *J. Intell. Wheelchair Syst.,* vol. 17, no. 3, pp. 243 – 264, 1996.

[6] S. Lee, "Use of infrared light reflecting landmarks for localization," *Ind. Rob:* An int' jnl., vol. 36, no. 2, pp: 138-145, 2009.

[7] D. A. Sanders, M. Langner, M., and G. E. Tewkesbury, "Improving wheelchair-driving using a sensor system to control wheelchair-veer and variable-switches as an alternative to digital-switches or joysticks *Ind Rob*: *An int' jnl,* vol 32, no. 2, pp. 157 – 167, 2010.

[8] M. Langner, "Effort Reduction and Collision Avoidance for Powered Wheelchairs: SCAD Assistive Mobility System," PhD. Dissertation, Dept. Mech. and Design Eng., Univ. of Portsmouth, Portsmouth, UK, 2012.

[9] D. A. Sanders and N. Bausch, "Improving Steering of a Powered Wheelchair Using an Expert System to Interpret Hand Tremor". *Proc of Intelligent Wheelchairics and Applications (Icira 2015), Pt Ii,* Portsmouth. UK, 2015, vol. 9245, pp. 460-471.

[10] D. A. Sanders, "Using self-reliance factors to decide how to share control between human powered wheelchair drivers and ultrasonic sensors," *IEEE T Neur. Sys. Reh.,* vol. 25, no. 8, pp. 1221-1229, 2017.

[11] D. A. Sanders et al., "A Rule-based Expert System to decide on direction and speed of a powered wheelchair," *Adv. Intell. Syst. Comput.,* vol. 868, Springer, pp. 822-832, 2019.

[12] M. Haddad *et al.,* "Combining multiple criteria decision making with vector manipulation to decide on the direction for a powered-wheelchair," *Adv. Intell. Syst. Comput.,* vol. 1037, Springer, pp. 680-693, 2019.

[13] M. J. Haddad *et al.,* "Intelligent HMI and Control for Steering a Powered Wheelchair Using a Raspberry Pi Microcomputer," in *Proc. 2020 IEEE 10th International Conference on Intelligent Systems,* Varna, Bulgaria, 2020, Accepted and in Press.

[14] D. Sanders, M. Langner, and G. Tewkesbury, "Improving wheelchair-driving using a sensor system to control wheelchair-veer and variable-switches as an alternative to digital-switches or joysticks", *Ind Rob: An int' jnl,* vol. 37, no. 2, pp. 151-167, 2010.

[15] S. Lee, "Use of infrared light reflecting landmarks for localization", *Ind Rob: An int' jnl,* vol. 36, no. 2, pp: 138-145, 2009.

[16] D. Sanders, and I. Stott, "A new prototype intelligent mobility system to assist powered wheelchair users", *Ind Rob: An int' jnl*, vol. 26, no. 6, pp. 466-475, 2009.

[17] J. Larsson, M. Broxvall, and A. Saffiotti, "Laser-based corridor detection for reactive Navigation", *Ind Rob: An int' jnl,* vol. 35, no. 1, pp 69-79, 2008.

[18] V. Milanes, J. Naranjo, and C. Gonzalez, "Autonomous vehicle based in cooperative GPS and inertial systems", *Robotica*, vol. 26, pp. 627-633, 2008.

[19] D. A. Sanders, "Controlling the direction of walkie type forklifts and pallet jacks on sloping ground". *Assem. Autom.,* vol. 28, no. 4, pp. 317-324, 2008.

[20] D. Sanders, "Recognizing shipbuilding parts using artificial neural networks and Fourier descriptors", *Proceedings of the Institution of Mechanical Engineers - Part B- Journal of Engineering Manufacture,* vol. 223, no. 3, pp. 337-342, 2009a.

[21] Y. C. Chang, and Y. Yamamoto, "On-line path planning strategy integrated with collision and dead-lock avoidance schemes for wheeled wheelchair in indoor environments", *Ind Rob: An int' jnl,* Vol. 35, no.5, pp. 421-434, 2008.

[22] D. A. Sanders *et al.,* "Tele-operator performance and their perception of system time lags when completing mobile robot tasks," in *Proc 9th Int Conf on Human Systems Interaction,* Portsmouth, UK, 2016, pp. 236-242.

[23] S. Hochreiter, and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735-1780, 1997.

[24] F. A. Gers, "Learning to forget: Continual prediction with LSTM," in *Proc. 9th International Conference on Artificial Neural Networks: ICANN '99,* Edinburgh, UK, 1999, pp. 850–855.

[25] K. Greff *et al.,* "LSTM: A Search Space Odyssey". *IEEE T Neur Net Lear.,* vol. 28, no. 10, pp. 2222- 2232, 2015.

[26] A. Graves *et al.,* "A Novel Connectionist System for Unconstrained Handwriting Recognition," *IEEE T Pattern Anal*., vol. 31, no. 5, pp. 855–868, May, 2009.

[27] U. Côté-Allard, et al., "Deep learning for electromyographic hand gesture signal classification using transfer learning", *IEEE T Neur. Sys. Reh.*, vol. 27, no. 4, pp. 760-771, 2019.

[28] M. Montemerlo, *et al.,* "Experiences with a mobile robotic guide for the elderly". *AAAI/IAAI*, pp.587-592, 2002.

[29] D. B. Argall, *et al.,* "A survey of robot learning from demonstration." Robotics and autonomous systems, vol. 57, no. 5, pp. 469-483, 2009.

[30] N. Ratliff,*et al.,* "CHOMP: Gradient optimization techniques for efficient motion planning," in *Proc. 2009 IEEE International Conference on Robotics and Automation,* Kobe, Japan, 2009, pp. 489-494.

[31] S. Tellex, *et al.,* "Understanding natural language commands for robotic navigation and mobile manipulation." In *Proc. Twenty-fifth AAAI conference on artificial intelligence,* .San Fransisco, California, USA, 2011, pp.1507-1514.

[32] D. A. Sanders, "The modification of pre-planned manipulator paths to improve the gross motions associated with the pick ad place task," *Robotica,* vol 13, pp 77-85, 1995.

[33] D. Sanders, "Viewpoint - Force sensing," *Ind Rob: An int' jnl,* vol 34, page 177. 2007.

[34] D. Sanders, "Comparing ability to complete simple tele-operated rescue or maintenance mobile-robot tasks with and without a sensor system," *Sensor Review,* vol 30, no. 1, pp: 40-50. 2010.

[35] D. Sanders *et al.*, "Introducing time-delays to analyse driver reaction times when using a powered wheelchair," in *Proc. 2020 IEEE 10th International Conference on Intelligent Systems*, Varna, Bulgaria, 2020, Accepted and in Press.

Malik J. Haddad was born in Amman, Jordan in 1984. He received a BSc in Electronic Engineering in 2006 (PSUT). He went on to postgraduate study and was awarded a distinction in MSc in Electrical and Computer Engineering in 2007 (NYIT). Finally, he completed a PhD in Decision Making applied to Management and Engineering in 2019 (University of Portsmouth).

Current research interests include decision-making, especially multiple criteria decision making, assistive technologies, obstacle avoidance, Artificial Intelligence, and Neural Networks.

Dr Haddad is a Member IEEE, Member IMechE, Member IET and a Fellow of the Jordan Engineers Association.

David A. Sanders MBE TD VR CEng was born in London, UK in 1958. He received two first degrees, a BA in Social Science and Mathematics in 1985 (OU), and a BSc in Engineering in 1984 (Portsmouth, UK). He went on to post graduate study and was awarded a DMS in Management Studies and then a MBA in Business Administration and Change Management in 2001 (OU). He completed a PhD in Automatic Path Planning with constraints in 1990 (CNAA, UK).

Since 2019 he has been the Professor of Systems Engineering and Reader in Knowledge Engineering in the Faculty of Technology at the University of Portsmouth. He is the author of three books, more than 250 articles, and several patents and inventions. Current and previous research interests include assistive technologies, sensor integration, robotics, AI, tele-operation, powered wheelchairs and obstacle avoidance.

Professor Sanders is a Fellow IET, Fellow IMechE, and Fellow HEA.