e-ISSN : 0976-5166
p-ISSN : 2231-3850

CONTROL OF AN AUTONOMOUS VEHICLE WITH OBSTACLES IDENTIFICATION AND COLLISION AVOIDANCE USING MULTI VIEW CONVOLUTIONAL NEURAL NETWORK

M.Karthikeyan[1] and S.Sathiamoorthy[2]

[1]Division of Computer and Information Science, Annamalai University, Chidambaram
[1]mkarthi82@gmail.com

[2] Tamil Virtual Academy, Chennai
Ks_sathia@yahoo.com

Abstract

Artificial Intelligence (AI) is inevitable in this era for automation requirements in all large scale industries like Automotive, Aerospace, Railways, Industrial automation, and Renewable energy industries. Among the AI techniques, deep learning algorithm with artificial neural network (ANN) receives greater attention on estimation and control requirements. In this paper, control of autonomous passenger vehicle using deep multi view convolutional neural network (CNN) for the identification of obstacles with 3 dimensional (3D) images of the same using Winograd Minimal filter algorithm (WMFA) has been presented. Authors also have clearly articulated the accuracy level difference between Machine learning (ML) algorithm, basic CNN algorithm and the proposed CNN algorithm in this paper for obstacle identification, collision avoidance and steering control. Most importantly, training of neural networks with multi view topology using Matlab/Simulink coding has been presented with the results. Real-time 3D images have been captured and compared with the stored and trained data. Output of trained CNNs have been captured and the results have been compared and discussed in this paper.

Key Words: Autonomous Vehicle, Neural Networks, CNN, Machine Learning and Deep Learning, WMFA

1. Introduction

Autonomous vehicle technology creates greater impact in automotive industry and the manufacturers are focusing on this technology from last decade. Although autonomous vehicles are not yet introduced in commercial market to greater extent, it has reached many milestones in development phase [1][2][3]. Even few research firms have done the trial run on the highways successfully in worldwide. Moreover, USA & Japan based agricultural vehicles manufacturers have introduced Autonomous tractors in the market which are mainly used for forming and cultivation [4]. Autonomous tractors have lesser complexities than the autonomous passenger cars and trucks as more external influencing factors such as unknown road conditions, obstacles, possibility of collision with front and rear vehicles, road friction, etc. Hence, more control & intelligence systems are required to achieve the complete features of autonomous passenger vehicles and on-road trucks [5][6]. In view of the above, authors have presented a novel algorithm to achieve the certain features of self-driven or autonomous passenger vehicles like Steering control, auto braking, tyre pressure monitoring and Collision avoidance system. Figure 1 shows the schematic diagram of Autonomous car with proposed inputs and desired outputs and which describes the scope of this paper as well. When the vehicle is in driving mode on the road, LiDAR (Light detection and Ranging) captures the images in front and rear side of the vehicle across the lane. These images captured as 3D images and processed through the artificial intelligence with additional inputs of tyre pressure and road conditions. Based on the obstacle position and conditions, steering angle of the vehicle & braking pressure values are determined then applied accordingly. This paper mainly focussed on obstacle identification and collision avoidance to protect the vehicle. This helps to keep the vehicle between the lane.

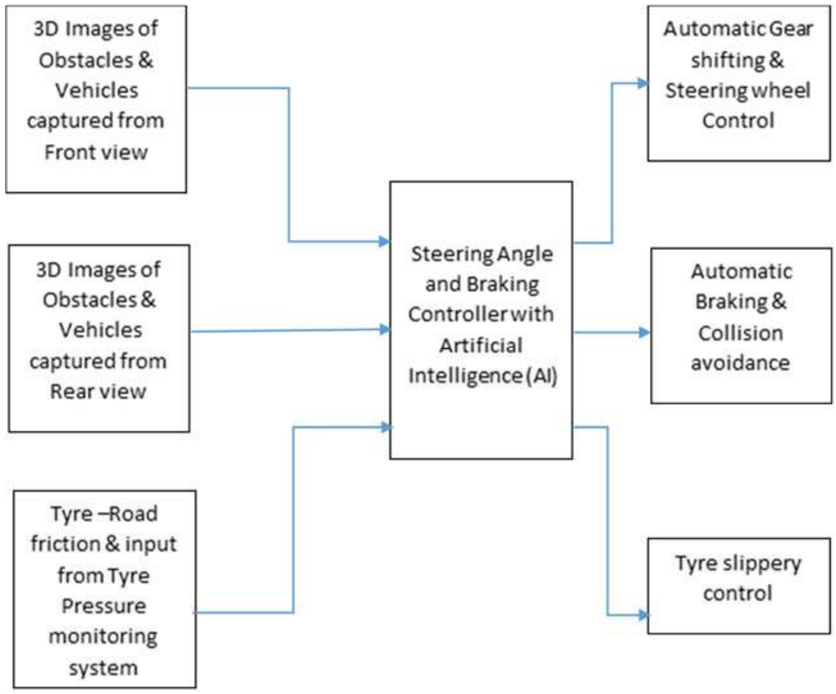


Figure 1 Schematic of Autonomous Vehicle

Cameras or LiDAR systems are used in most of the vehicles to capture the actual images in order to estimate the steering angle. 3D images have been captured using LiDAR in this work and analysis has been presented in this paper. LiDAR sensor provides the direct 3D representation of obstacles including the surrounding information in the form of 3D clouds [7]. Precision of the 3D measurement depends on the calibration level of LiDAR with 2 cm accuracy level of 360 degrees filed view. In most of the work in the past, LiDARs had been used in 2D based CNN approach and LiDAR is the well proven sensor as well [8].

2. Machine Learning (ML) Algorithm Technique

Machine learning is an applied statistics algorithm and is used to find the available data and predict the output. Machine Learning algorithm is classified as supervised and unsupervised algorithm [9]. In any application, basically if the data is not available and available data will be used to find the output. Unsupervised learning technique is used only to predict the output with available data. If any model contains only the compound data without any output, then unsupervised algorithm will be used. Supervised learning technique is used to review output with available input and output data. Final clustered data will be obtained by direct unsupervised learning. Classified and regression techniques are the types of supervised learning. In many nonlinear applications, unsupervised learning is mainly used to predict the output [10][11]. In most of the passenger cars, all the data available including obstacle images, vehicle speed, engine temperature, steering angle and Tyre pressure data. Predictive models are developed by both supervised and unsupervised learning techniques. In this paper, authors have processed the captured 3D images through supervised algorithm as well. The main scope of this paper is to compare the performance of autonomous vehicle while crossing different obstacles using different image processing algorithm. Obstacle avoidance and Lane keeping are the main requirements of an autonomous vehicle in order to achieve the safe drive. Various researchers have applied many algorithms to accomplish these features [12][13][14]. Figure 2 shows the schematic of Autonomous car controller with Machine learning algorithm for obstacle avoidance and Lane Keeping. Actual images of obstacles have been given as inputs to the network for classification. So many machine learning algorithms have been tested for classification in the autonomous car field such as Reinforcement learning, Support Vector Machine (SVM), neural networks, etc. Reinforcement algorithm is being used for identification of objects in order to avoid the collision between the vehicles and objects whereas SVM technique is used to keep the vehicle between the lane on the highways [32].

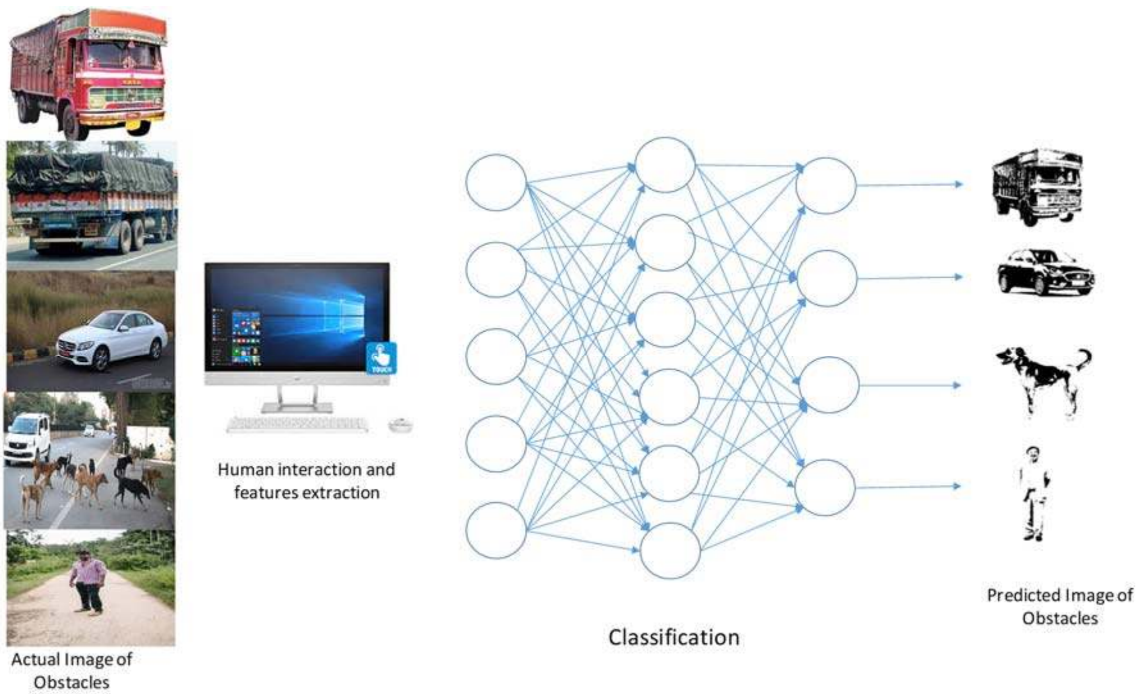


Figure 2 Network structure of Machine Learning Algorithm

Figure 3 shows the schematic of autonomous car using machine learning algorithm. Reinforcement Learning has been used in this schematic in association with Proportional Integral Derivative (PID) control algorithm in order to control the steering angle based on the image inputs.

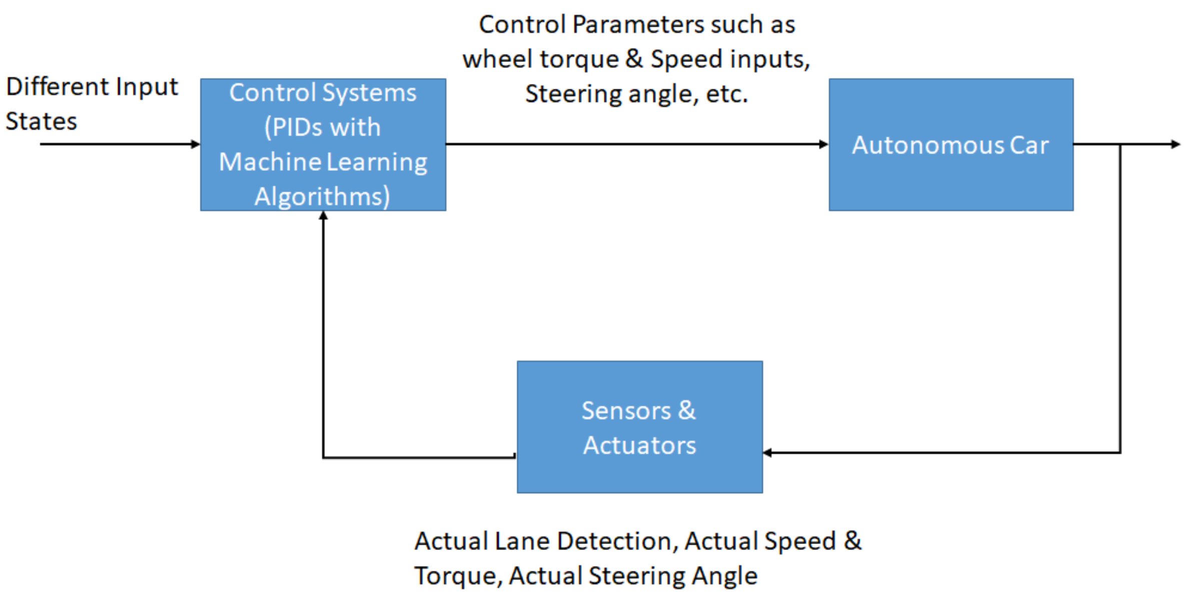


Figure 3 Schematic of Supervised Learning based controller for Autonomous car

Figure 4a shows the flowchart for training the data and deployment using reinforcement algorithm. Road conditions with obstructions and vehicle data includes speed, fuel conditions, temperature, tyre pressure, etc. have been taken as input for pre-processing. Then captured data got into analysis, training and learning.

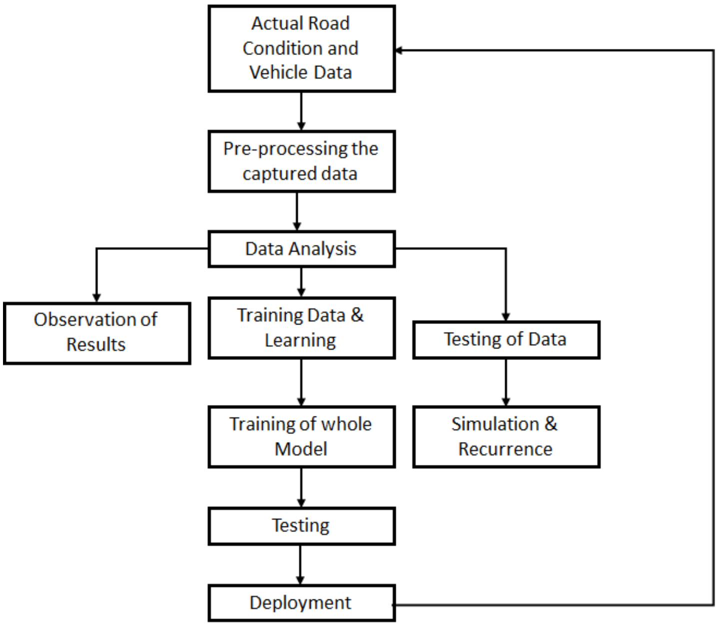


Figure 4a Flowchart of Data training and deployment

In order to achieve the goal using ML algorithm, authors have captured all the possible combinations of the real time obstacle images and supervised learning algorithm has been applied. Real-time images considered in this paper are front and rear view of different vehicles, flying birds, speed breakers, crossing animals and road signals. All these images have been labelled with their respective categories. Proposed autonomous vehicle controller has been modelled in this paper contains the actual images of all these categories and features have been captured carefully. Figure 4b shows the results captured using Matlab/Simulink, by Mathworks© [33] with Reinforcement algorithm for an autonomous vehicle. All of the results captured are given in the picture itself and contains the episode information, average results, training options and final results. Graphs shown in figure 4b depict episode reward and average rewards of training the data. K-Nearest Neighbour (KNN) algorithm is also tested and the result is shown in figure 4c.

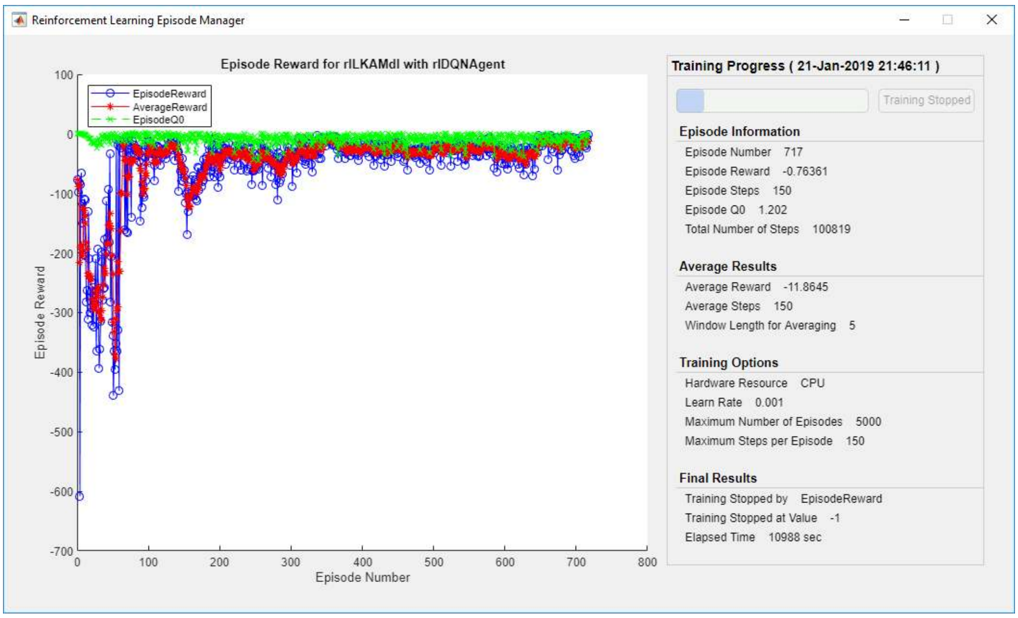
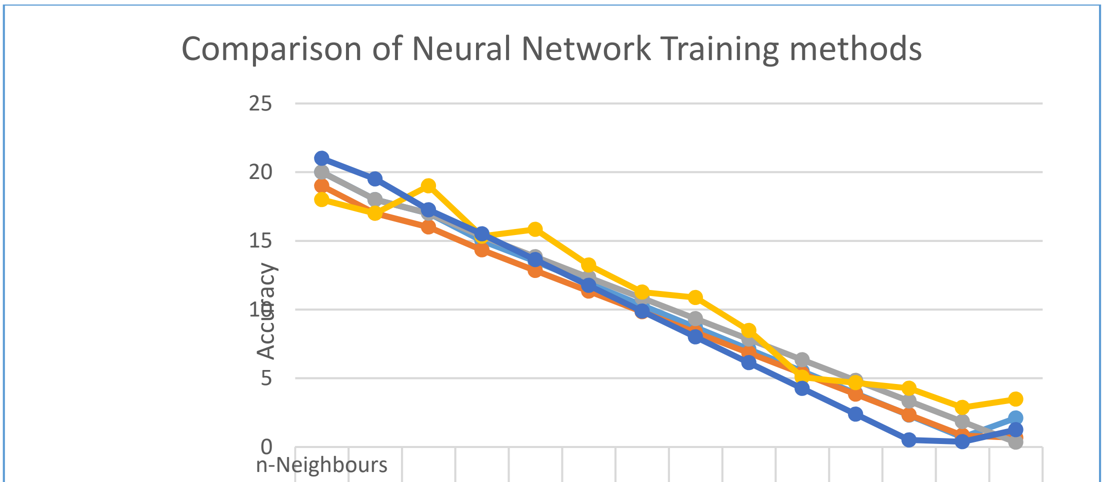


Figure 4b: Reinforcement algorithm training pattern using Matlab/Simulink©



Train	20	18	17	15	13.5	11.9	10.3	8.7	7.1	5.5	3.9	2.3	0.7	2.1
Validation	19	17	16	14.33	12.83	11.33	9.833	8.333	6.833	5.333	3.833	2.333	0.833	0.667
Train-Scaling	20	18	17	15.33	13.83	12.33	10.83	9.333	7.833	6.333	4.833	3.333	1.833	0.333
Validation-Scaling	18	17	19	15.33	15.83	13.23	11.27	10.87	8.467	5.067	4.667	4.267	2.867	3.467
Train-Scaling	21	19.5	17.25	15.5	13.63	11.75	9.875	8	6.125	4.25	2.375	0.5	0.375	1.25

Figure 4c KNN algorithm training pattern using Matlab/Simulink ®

3. Deep Learning (DL) Algorithm Techniques

Deep Learning Algorithm is differentiated from conventional Machine learning algorithm with respect to the prediction error [15]. DL algorithm analyses the each and every features of the actual image and train the neural networks in order to predict the actual image and appropriate action will be taken accordingly [16]. Prediction accuracy is much better by using deep learning algorithm than machine learning algorithm. Because, in machine learning algorithm, human interaction is required to extract the information from the image or data to predict the model or class and this process will affect the prediction accuracy and cause more error. Whereas in deep learning algorithm technique, both information capture and prediction are well taken care by network itself in multiple stages and prediction error will be drastically reduced. Deep learning neural networks are capturing the meaningful information from the raw data and doing the prediction [17].

4. Proposed Convolutional Neural Network Algorithm

In general, a Neural network has multiple layers and one of the layers applies the mathematical operation of convolution is known as Convolutional neural network [18-23] [34]. Convolutional neural network has three important layers such as convolution, pooling and fully connected layers as shown in figure 4. Features of an image is extracted and processed through multiple convolutions using mathematical functions and pooled then processed through fully connected layer also known as dense layer. Authors have presented multi-view 3D convolutional algorithm running on Graphics Processing Unit (GPU) [24] to control the autonomous vehicle in this paper.

Equation 1 shows the function of general Convolutional neural network with forward inference.

$$u = conv(x, w) \qquad (1)$$

Where u= output of the Convolutional Layer

x = information extracted from image

w = filter to apply on the image

Other important function in CNN is activation function which is applied on the features extracted from image. This helps to process the information for pooling and mapping. Among various activation functions, Rectified Linear Unit (ReLU) [25] has been used in this paper. In this paper, description of 3D images and videos extracted and classified by convolution network has been presented in detail with the proposed algorithm and data points. In figure 5, 3D image of vehicle obstruction in front of autonomous vehicle has been taken for the processing and the feature extraction has been taken place. Convolutional steps then be started with the proposed algorithm in order to speed up the convolutional process without increasing the memory size. In this approach, every neuron in the input layer is connected to every neuron at the output layer. This fully connected layer is used to classify the features of the images captured into various classes based on the training data set.

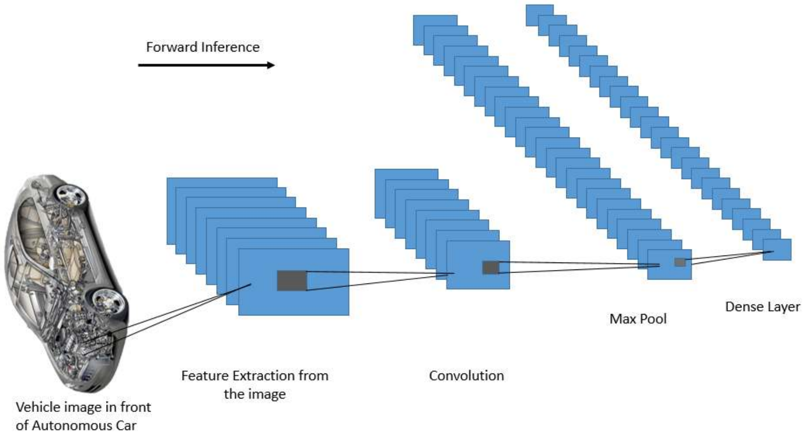


Figure 5 Convolutional Neural Network Algorithm

Figure 6 shows the proposed convolutional neural network which has real time images or dataset being processed with multi-view 3D CNN. In this algorithm, input x has different dimension of the convolutional features map includes length, height, width, number of channels and batch size which used to process the 3D images with maximum accuracy. This can be written as given in equation (2)

$$x = [L\ M\ N\ K\ S] \tag{2}$$

Where x = inputs

L=Length of the convolutional features map

M = Map height

N = Map width

K = number of convolutional channels

S= convolutional batch size

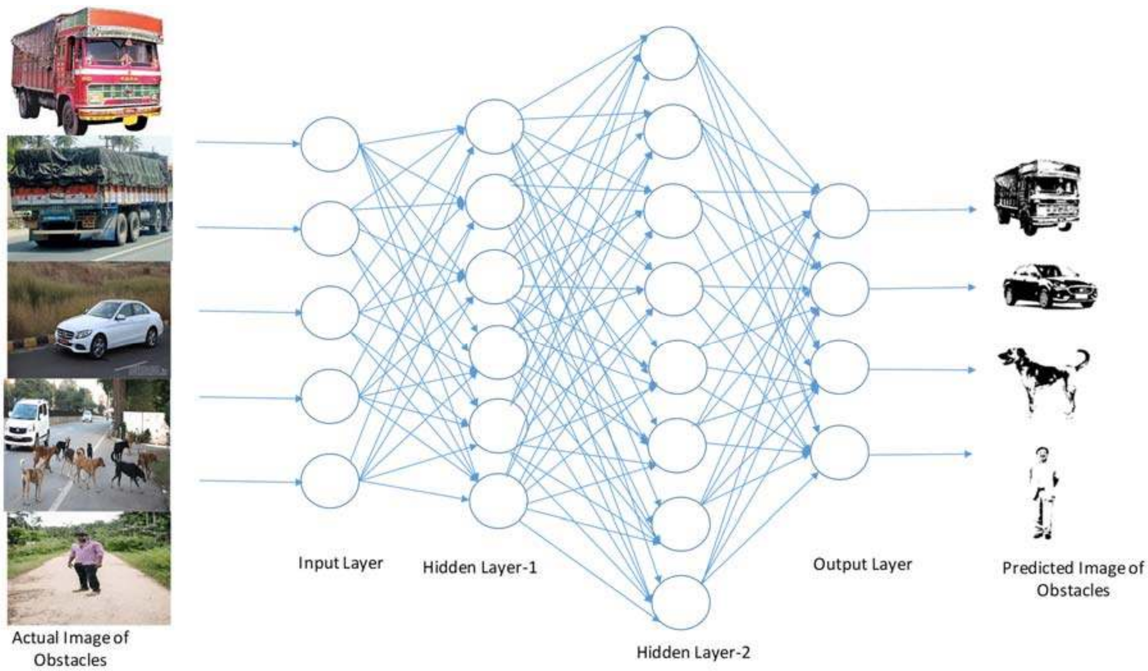


Figure 6 Function of proposed algorithm

2D convolutional neural networks have been applied in autonomous vehicles so far and are being analysed in many research articles [26-31]. In 2D convolution, width and the height of the images have been fixed by kernels and this would be inclined with the width and height of the input features of reference maps.

Output of 3D convolutional layer is given in equation (3). This output is the result of the convolution operation at kth channel. 'I' represents the input feature and 'F' represents the filters.

$$Y_{i,k,x,y,z} = \sum_{C=0}^{C-1} \sum_{t=0}^{T-1} \sum_{r=0}^{R-1} \sum_{s=0}^{S-1} \mathrm{I}_{i,c,x,y+r,z+s}\, \mathrm{F}_{k,t,r,s,c'} \tag{3}$$

Equation 3 represents the straight forward convolution with intensive computability. Authors have implemented Winograd Minimal filter algorithm (WMFA) for 3D Convolution application in order to find the obstacles and to turn the steering angle of Autonomous car. 3D WMFA convolutional algorithm needs more calculation time when compare to 2D WMFA algorithm in order to complete the iteration to estimate the output. However, 3D WMFA is faster than the conventional 3D convolution in reprocessing the input images. WMFA finds the output with a frame size of 'm' each time. F (m, r) represents the output frame where 'r' is the filter size. In Convolution network theory, 'mxr' multiplications are required to compute F (m, r), but number of convolutions can be reduced by using the equation (4)

$$F(m,r) = \begin{bmatrix} i_0 & i_1 & i_2 \\ i_1 & i_2 & i_3 \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} m_1 + m_2 + m_3 \\ m_2 - m_3 + m_4 \end{bmatrix} \tag{4}$$

Where,

$m_1 = (i_0 - i_2)\, f_0$

$m_2 = (i_1 + i_2)\, \dfrac{f_0 + f_1 + f_2}{2}$

$m_3 = (i_2 - i_1)\, \dfrac{f_0 - f_1 + f_2}{2}$

$m_4 = (i_1 - i_3)\, f_2$

$i_0, i_1, i_2, i_3 = input\ image\ frames$

$f_0, f_1, f_2 = filter\ frames$

In order to compute the size of the image, 3D WFMA algorithm is used where multiplication of tile size and filter size calculates the convolution. However, number of multiplication is being reduced to minimize the convolutions in this paper.

5. Characteristics of Real-Time 3D images captured

While autonomous vehicle is on the road, many obstacles including living things may cross the road and vehicle should control the speed automatically after sensing the obstacles. Most commonly, front vehicles, humans, animals and speed breaker would be the obstacles and which may hit the vehicle if it is not controlled. Hence, real time images of those obstacles have been considered as dataset and to derive the map. Those pictures were given as inputs to multi view 3D convolutional neural networks wherein convolution, max pooling and dense process functions were being carried out.

6. Training of Neural Networks

Stage by stage training has been applied with the proposed algorithm in this paper. As described above, captured 2D images have been converted into 3D multiple view images with the help of 3D convolutional network. Features of the captured images have here been triggered by an activation function called Rectifier Linear Unit (ReLU) and which gives more accuracy of the prediction. In general, there are 2 methods for deep learning more effective as below:

- 3D WMFA
- cuDNN

3D WMFA is used to speed up the 3D convolutions without increase the memory compare to cuDNN and this is also one of the reasons to choose the 3D WMFA from 3D convolutional library. Table-1 shows the data captured from this work during the execution of this deep learning algorithm.

Table 1: Data Captured from Proposed Algorithm Training

Epoch	Iteration	Time Elapsed	Mini-batch (Loss)	Mini-batch (Accuracy)	Base Learning (Rate)
2	50	0.45	2.2301	47.66%	0.010000
3	100	0.88	0.988	75.00%	0.010000
4	150	1.31	0.5558	82.03%	0.010000
6	200	1.75	0.4022	89.06%	0.010000
7	250	2.18	0.375	88.28%	0.010000
8	300	2.613	0.3368	91.41%	0.010000
10	350	3.046	0.2589	96.09%	0.010000
11	400	3.479	0.1396	98.44%	0.010000
12	450	3.912	0.1802	96.09%	0.010000
14	500	4.345	0.0892	99.22%	0.010000
15	550	4.778	0.1211	96.88%	0.010000
16	600	5.211	0.0961	98.44%	0.010000
18	650	5.644	0.0856	99.22%	0.010000
19	700	6.077	0.0651	100.00%	0.010000
20	750	6.51	0.0582	98.44%	0.010000
22	800	6.943	0.0808	98.44%	0.010000
23	850	7.376	0.0521	99.22%	0.010000
24	900	7.809	0.0248	100.00%	0.010000
25	950	8.242	0.0241	100.00%	0.010000
27	1000	8.675	0.0253	100.00%	0.010000
28	1050	9.07	0.026	100.00%	0.010000
29	1100	9.539	0.0246	100.00%	0.010000

7. Results and Discussions

Four data sets have been taken for training and validation and the results have been presented in this paper. Datasets and the related information have been given in Table-1. Figure 7 shows the training status of CNN for classification of the features from the input images. Trained data set is compared with the reference data set or test set to predict the output from the classification layer. From the figure 7, it is observed that success rate for the classification is about 99.4%. It is clearly evident that training has been done very well and trained data is almost in line with the test data. Figure 7 also shows that loss of classification of the features or difference between the trained data set and test data set and the value is around 0.25 only. Hence from this result, it is clear that proposed algorithm is very effective and matured.

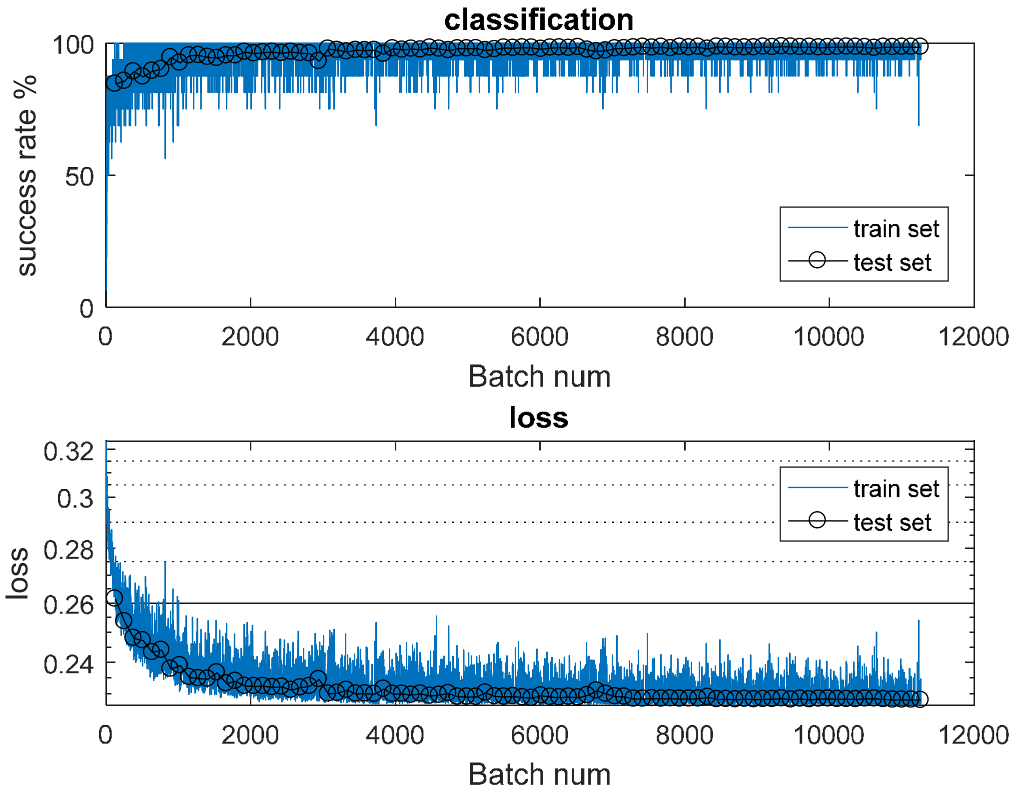


Figure 7. Training Status

Figure 7 shows the representation of success rate for the prediction of features from different dataset captured as input images. Different obstructions in front of autonomous vehicle to be captured precisely and the success rate to be measured. Based on the features of obstruction in front of vehicle, autonomous vehicle shall change its steering angle in order to avoid the collision. Hence, feature prediction success rate plays vital role to avoid the collision while autonomous vehicle in driving mode on the road. In figure 8, it is evident that most of the features have been extracted with 99.4% accuracy in order to predict the position of obstacles on the road.

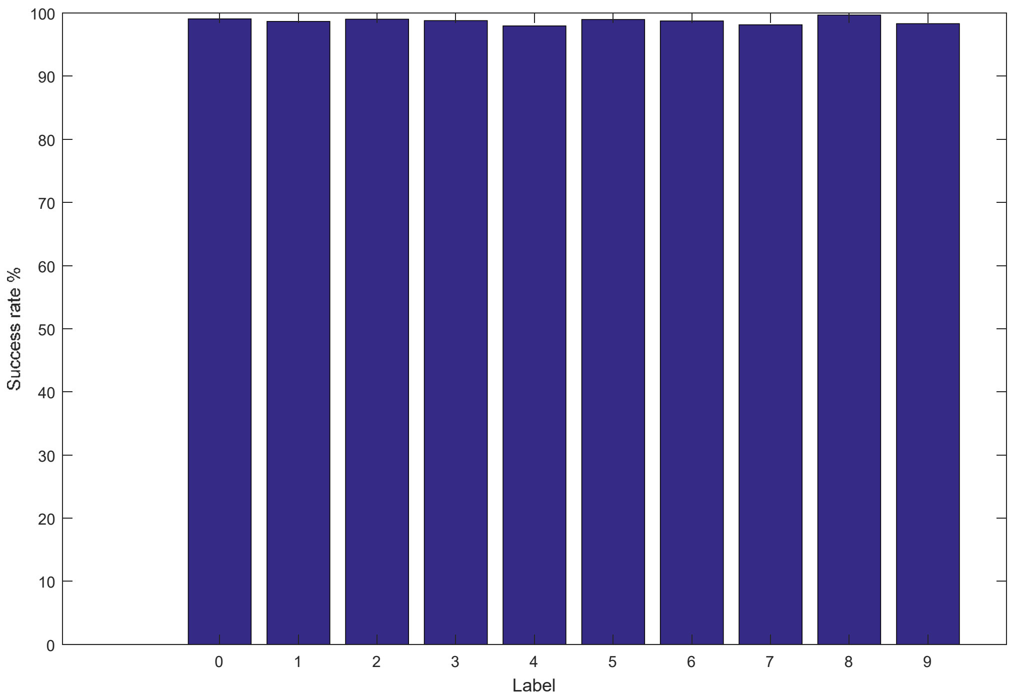


Figure 8. Success rate

Moreover, more training samples have also been taken during the experiment which minimize the prediction error and increased the success rate. Authors have also tested with various convolutional filter size and captured the results that when dimensions of convolutional filter increases, prediction error is also increased which directly impacts on accuracy and success rate. Hence, optimization of filter size or dimension is also equally important to achieve the minimum prediction error. Figure 9 shows the surface plot of CNN estimation and this plot is between the different labels data set, percentage of success rate and the estimated output. All these graphs are captured from Matlab/Simulink modelling platform. In this research work, these predictions used activation of inner layers of CNN which was enabled by using CUDA enabled GPU with computation capability. CUDA enabled GPU is a parallel computing platform which is speeding up the computations especially in highly complex applications like autonomous cars, aerospace, automated industrial applications, etc., In this GPU, power of GPU is being harnessed for increasing the speed of the computations. Figure 10 shows the linearized input to convolutional filter which shows the input distributions across the layer.

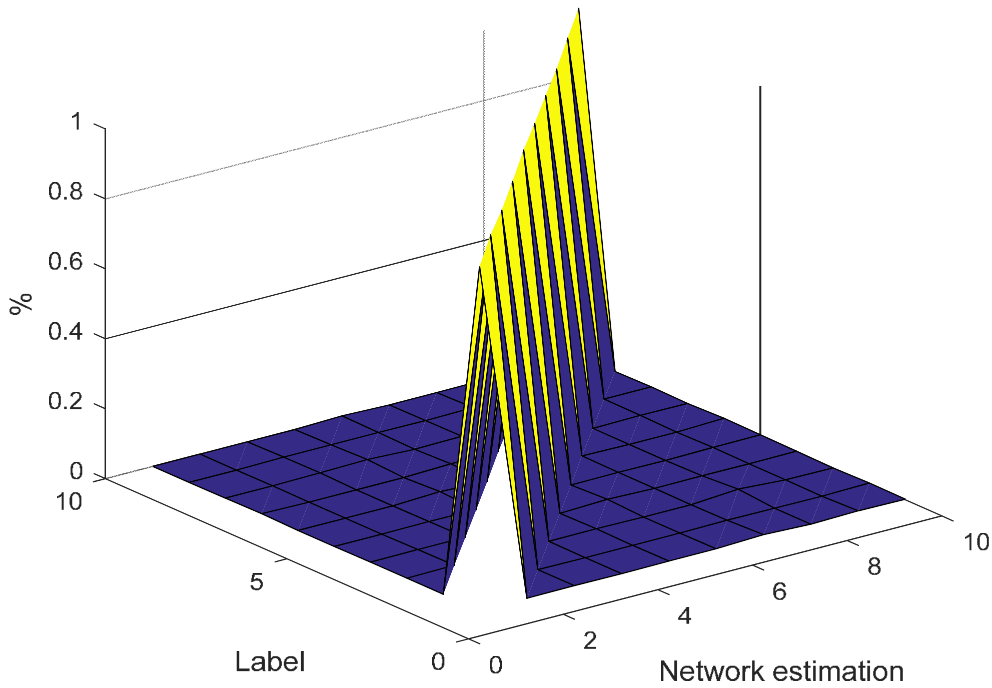


Figure 9. Network estimation

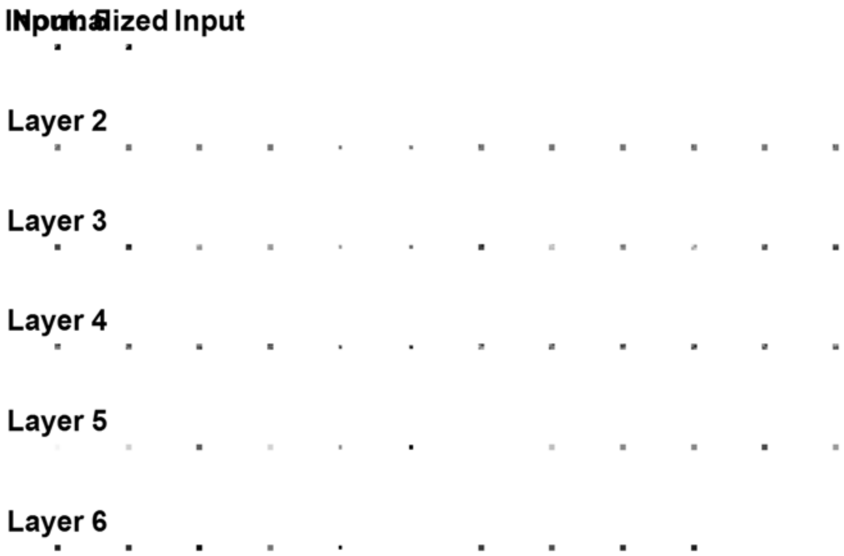


Figure 10. Network filter

Figure 11 shows the confusion matrix with estimates and actual values. There are 10 classes are taken into consideration as below:

1. Vehicle_change_lane
2. Vehicle_stop
3. Human_cross
4. Dog_cross
5. Vehicle_turn_right
6. Vehicle_turn_left
7. Truck_cross
8. Car_cross
9. Vehicle_reverse
10. Vehicle_run

All the above 10 classes are considered as output which needs to be compared with target classes in the same category. Total accuracy of the classification is 99.4% as shown in confusion matrix.

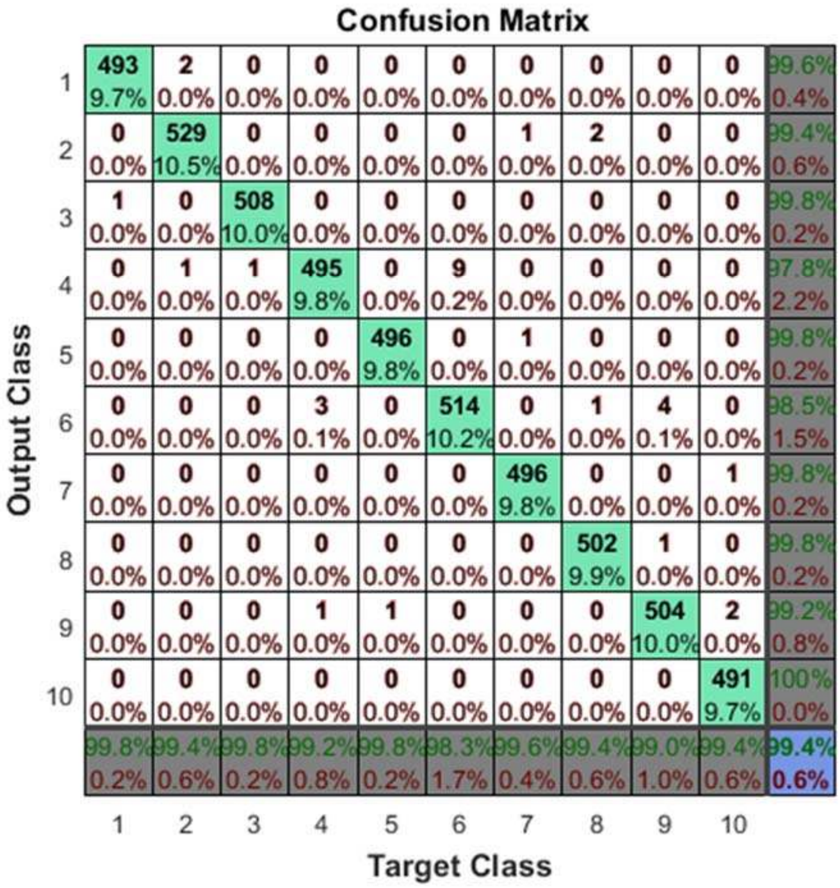


Figure 11. Confusion Matrix

Figures 12 and 13 show the simulator results which show the vehicle movement before and after applying the proposed algorithm. Figure 12 shows the simulator output of Autonomous car on the path in driving mode. This path is designed in such a way that it has lane, left and right turns with obstacles. In figure 12(a), vehicle turns left and hit the object without object classification algorithm. Vehicle tried to maintain within the lane but at the same time obstacle crossover and hit it without any control. In figure 12(b), with the proposed classification and proposed algorithm, vehicle took left properly and went towards straight without hitting the object.

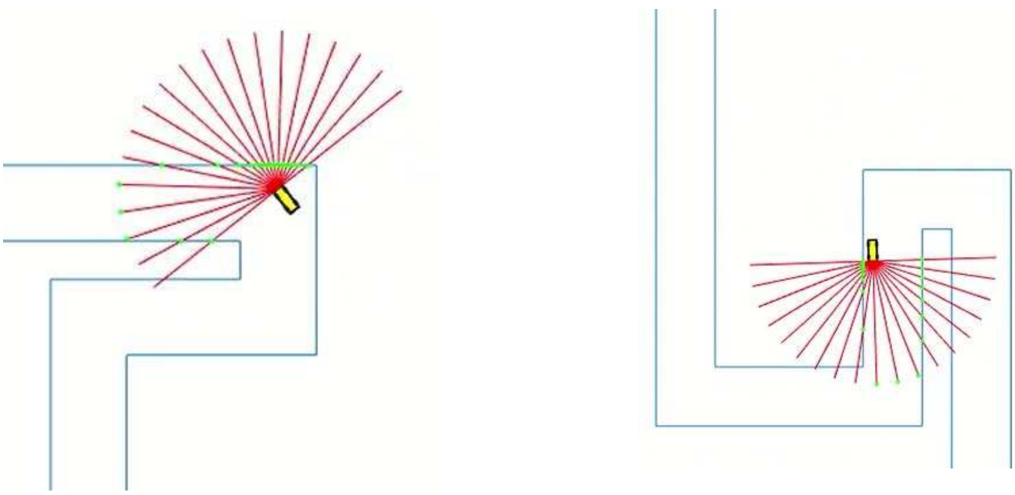


Figure 12 (a) Before Identifying obstacle and Collision (b) After Identifying obstacle and Collision avoidance using proposed algorithm

Figure 13 shows the graph of training the actual set data within the stipulated training period with and without proposed algorithm.

e-ISSN : 0976-5166
p-ISSN : 2231-3850

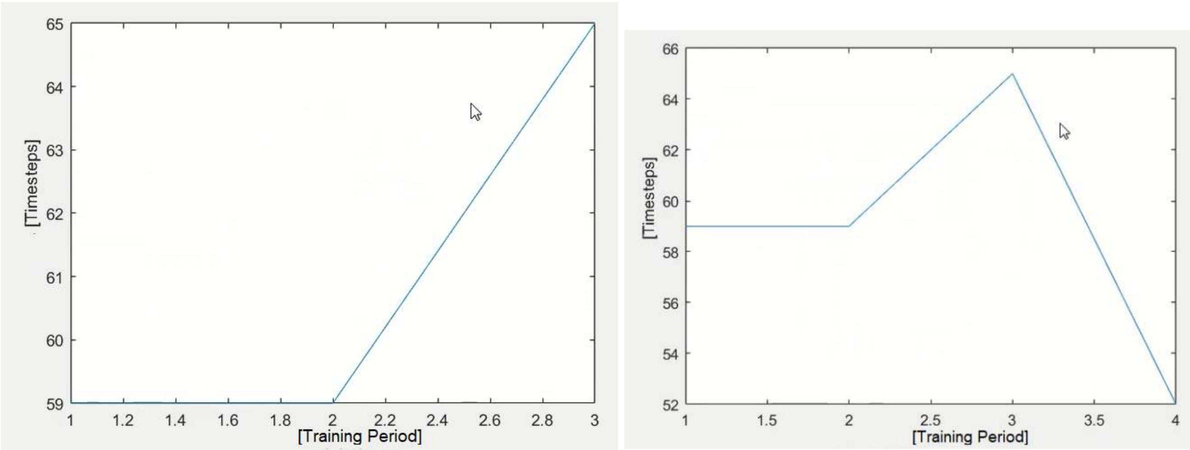


Figure 13 (a) CNN Training data without (b) With Proposed Algorithm

Matlab 2018b version has been used to write the M-script and develop the Simulink model to achieve the results presented in this paper. Figure 14 shows the proposed CNN algorithm results with the training accuracy which is close to 99%. Figure 15 shows the error measured during the training of CNN with proposed algorithm.

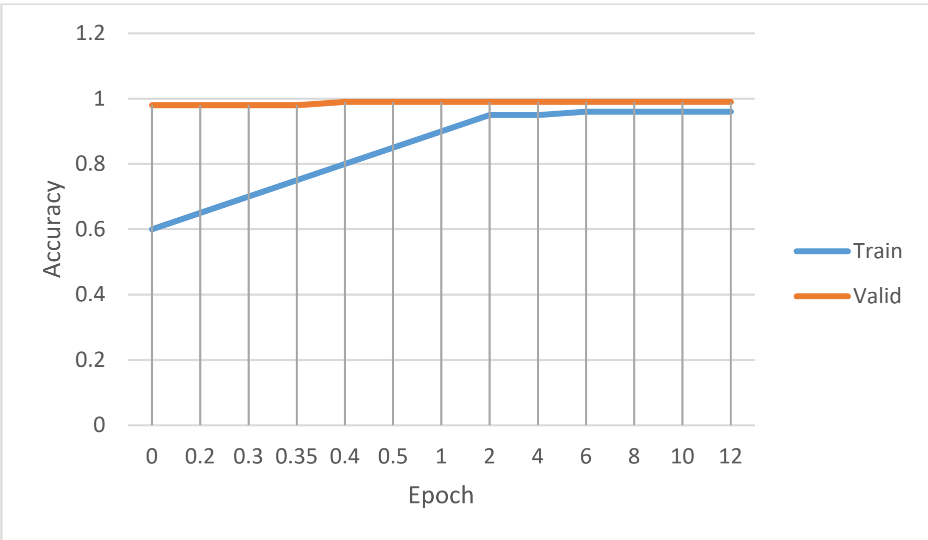


Figure 14 Proposed CNN algorithm training accuracy

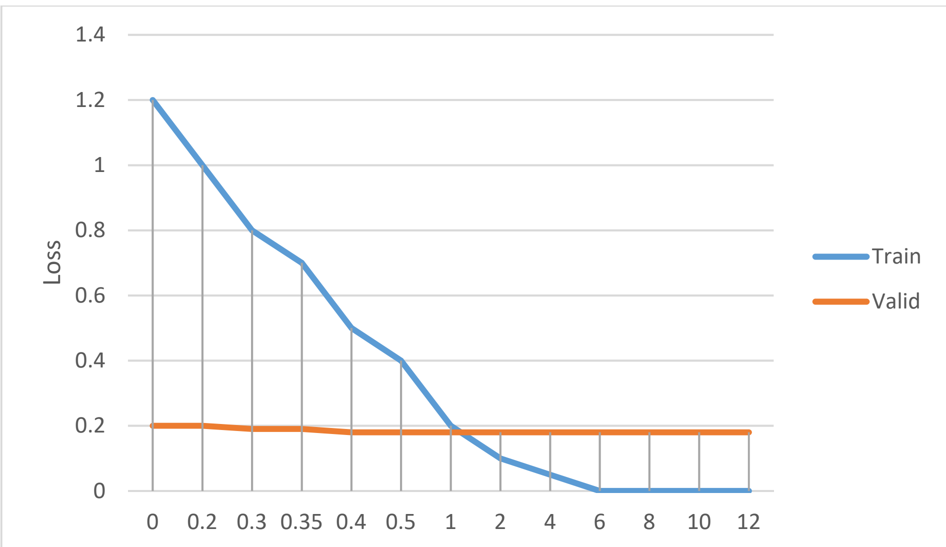


Figure 15 Error occurred during proposed CNN algorithm training

In this paper, performance of both Machine Learning and Deep Learning algorithm has been presented and the results have been discussed. KNN is the best ML algorithm for the test dataset in this paper providing the accuracy of 90.6% with the standard variation of 5.94 from the cross validation analysis based on 10 times. on the other hand, for DL architectures, CNN obtained the best accuracy with 94.43%.

8. Conclusion

The main objective of this paper was to prove the capability of 3D CNN with Winograd Minimal filter algorithm for controlling the autonomous vehicle. The first part of the work was to conclude what type of network to use. After doing the literature study, it became clear that multi-view convolutional neural network was the obvious choice. The second part of this work shows that using deep learning for solving the problem is possible. It is however not completely successful at generalizing and it has problems when the obstacles are moved around. However, deep learning shows good potential to solving the problem with greater success than what the collected results shows in this work. Thus, the network was evaluated on a larger dataset in order to validate the 3D multi view CNN with WMFA algorithm. In this paper, the dataset collected were tested with both Machine learning algorithm and deep learning algorithm and found that deep learning algorithm is more effective. Presented results in this paper demonstrated the effectiveness of the proposed algorithm.

References

[1] Bernhart, Wolfgang, and Marc Winterhoff. "Autonomous driving: Disruptive innovation that promises to change the automotive industry as we know it." In *Energy Consumption and Autonomous Driving*, pp. 3-10. Springer, Cham, 2016.
[2] Wilfong, Gordon T. "Motion planning for an autonomous vehicle." In Autonomous Robot Vehicles, pp. 391-395. Springer, New York, NY, 1990.
[3] Fagnant, Daniel J., and Kara Kockelman. "Preparing a nation for autonomous vehicles: opportunities, barriers and policy recommendations." Transportation Research Part A: Policy and Practice 77 (2015): 167-181.
[4] Johnson, David A., David J. Naffin, Jeffrey S. Puhalla, Julian Sanchez, and Carl K. Wellington. "Development and implementation of a team of robotic tractors for autonomous peat moss harvesting." *Journal of Field Robotics* 26, no. 6-7 (2009): 549-571.
[5] Trepagnier, Paul Gerard, Jorge Emilio Nagel, Powell Mcvay Kinney, Matthew Taylor Dooner, Bruce Mackie Wilson, Carl Reimers Schneider Jr, and Keith Brian Goeller. "Navigation and control system for autonomous vehicles." U.S. Patent 8,050,863, issued November 1, 2011.
[6] Cheng, Hong. Autonomous intelligent vehicles: theory, algorithms, and implementation. Springer Science & Business Media, 2011
[7] B. Li, T. Zhang, and T. Xia. Vehicle detection from 3D Lidar using fully convolutional network. In Robotics: Science and Systems, 2016.
[8] M. Z. Zia, M. Stark, B. Schiele, and K. Schindler. Detailed 3D representations for object recognition and modeling. PAMI, 2013
[9] Christopher M. Bishop. Pattern recognition and machine learning. Springer,2006. ISBN 9780387310732.
[10] MELLO, RODRIGO F., and Moacir Antonelli Ponti. Machine learning: a practical approach on the statistical learning theory. Springer, 2018.
[11] Michels, Jeff, Ashutosh Saxena, and Andrew Y. Ng. "High speed obstacle avoidance using monocular vision and reinforcement learning." In Proceedings of the 22nd international conference on Machine learning, pp. 593-600. 2005.
[12] Karthikeyan, M., S. Sathiamoorthy, and M. Vasudevan. "Adaptive Neuro Fuzzy Inference System Based Obstacle Avoidance System for Autonomous Vehicle." In International Conference on Innovative Data Communication Technologies and Application, pp. 118-126. Springer, Cham, 2019.
[13] Al-Zaher, Tamer S. Abd, Amged M. Bayoumy, Al-Hossein M. Sharaf, and Yehia H. Hossam El-din. "Lane tracking and obstacle avoidance for Autonomous Ground Vehicles." In *2012 9th France-Japan & 7th Europe-Asia Congress on Mechatronics (MECATRONICS)/13th Int'l Workshop on Research and Education in Mechatronics (REM)*, pp. 264-271. IEEE, 2012.
[14] Kunchev, Voemir, Lakhmi Jain, Vladimir Ivancevic, and Anthony Finn. "Path planning and obstacle avoidance for autonomous mobile robots: A review." In International Conference on Knowledge-Based and Intelligent Information and Engineering Systems, pp. 537-544. Springer, Berlin, Heidelberg, 2006.
[15] Bechtel, Michael G., Elise McEllhiney, Minje Kim, and Heechul Yun. "Deeppicar: A low-cost deep neural network-based autonomous car." In 2018 IEEE 24th International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA), pp. 11-21. IEEE, 2018.
[16] Liu, Shuying, and Weihong Deng. "Very deep convolutional neural network based image classification using small training sample size." In 2015 3rd IAPR Asian conference on pattern recognition (ACPR), pp. 730-734. IEEE, 2015.
[17] Yoshua Bengio. Practical recommendations for gradient-based training of deep architectures. In Neural networks: Tricks of the trade, pages 437–478. Springer, 2012.
[18] Lee, Donghan, Youngwook Paul Kwon, Sara McMains, and J. Karl Hedrick. "Convolution neural network-based lane change intention prediction of surrounding vehicles for acc." In 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pp. 1-6. IEEE, 2017.
[19] B.-S. Hua, M.-K. Tran, and S.-K. Yeung. Pointwise convolutional neural networks. In CVPR, pages 974–993, 2018.
[20] Qiang Lan, Zelong Wang, et. Al., 'High Performance Implementation of 3D Convolutional Neural Networks on a GPU' Computation Intelligence and Neuroscience, 2017, 1-8.
[21] Guixia Kang, Kui Liu1, Beibei Hou, Ningbo Zhang, '3D multi-view convolutional neural networks for lung nodule classification', PLOS ONE | https://doi.org/10.1371/journal.pone.0188290, November 16, 2017, 1-21.
[22] J. Fan, W. Xu, Y. Wu, and Y. Gong, "Human tracking using convolutional neural networks," IEEE Transactions on Neural Networks, vol. 21, no. 10, pp. 1610–1623, 2010.
[23] Y. Feng, Z. Zhang, X. Zhao, R. Ji, and Y. Gao. GVCNN: Group-view convolutional neural networks for 3D shape recognition. In CVPR, pages 264–272, 2018.
[24] Strigl, Daniel, Klaus Kofler, and Stefan Podlipnig. "Performance and scalability of GPU-based convolutional neural networks." In 2010 18th Euromicro Conference on Parallel, Distributed and Network-based Processing, pp. 317-324. IEEE, 2010.
[25] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In ICML, pages 807–814, 2010
[26] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In NIPS, 2015

[27] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner,Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, UrsMuller, Jiakai Zhang, et al. 'End to end learning for self-driving cars' arXiv preprint arXiv:1604.07316, 2016.
[28] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. 'Learning long-term dependencies with gradient descent is difficult'. IEEE transactions on neural networks, 5(2):157–166, 1994.
[29] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In CVPR09, 2009.
[30] Alippi, Cesare, Cosimo de Russis, and Vincenzo Piuri. "A neural-network based control solution to air-fuel ratio control for automotive fuel-injection systems." IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 33, no. 2 (2003): 259-268.
[31] François Chollet et al. Keras. https://github.com/fchollet/keras, 2015.
[32] Karthikeyan, M., S. Sathiamoorthy, and M. Vasudevan. "Lane Keep Assist System for an Autonomous Vehicle Using Support Vector Machine Learning Algorithm." In International Conference on Innovative Data Communication Technologies and Application, pp. 101-108. Springer, Cham, 2019.
[33] www.mathworks.com
[34] G. Mukhtar, S. Farhan, "Convolutional Neural Network Based Prediction of Conversion from Mild Cognitive Impairment to Alzheimer's Disease: A Technique using Hippocampus Extracted from MRI," Advances in Electrical and Computer Engineering, vol.20, no.2, pp.113-122, 2020, doi:10.4316/AECE.2020.02013
[35] A.-V. Vladuta, M. L. Pura, I. Bica, "MAC Protocol for Data Gathering in Wireless Sensor Networks with the Aid of Unmanned Aerial Vehicles," Advances in Electrical and Computer Engineering, vol.16, no.2, pp.51-56, 2016, doi:10.4316/AECE.2016.02007.