ICIC Express Letters
Volume **16**, Number **1**, January **2022**

 ISSN 1881-803X
pp. **17–24**

TRAFFIC COLLISION WARNING USING DEEP LEARNING MODELS

Tuan Linh Dang[1], Thuy Hang Nguyen[1], Gia Tuyen Nguyen[2] and Thang Cao[2]

[1]School of Information and Communications Technology
Hanoi University of Science and Technology
No. 1, Dai Co Viet Road, Hanoi 100000, Vietnam
linhdt@soict.hust.edu.vn; hang.nt183523@sis.hust.edu.vn

[2]Machine Imagination Technology Corporation (MITECH)
3-7-87 Koyanagi, Fuchu, Tokyo 183-0013, Japan
{ tuyen; cao }@mitech.jp

Received May 2021; accepted August 2021

Abstract. *This paper investigates a traffic collision warning system that consists of two modules. The first module tracks the movements of pedestrians and vehicles presented in our previous paper. This manuscript focuses on the second module that forecasts the location of vehicles and pedestrians to predict the collisions. Three different models were employed called One-to-One Long Short-Term Memory (LSTM), Many-to-One LSTM, and neural network, respectively. Experimental results showed that the Many-to-One LSTM model could be a solution for the second module of the traffic collision warning system.*

Keywords: LSTM, NN, Time series forecasting, Traffic collision

1. **Introduction.** According to the World Health Organization, traffic safety is a global problem. Each year, around 1.35 million people die, and from 20 to 50 million people are affected by traffic accidents globally which also makes a loss of $1500 billion (accounting for 2.5% of gross domestic product, GDP, global) [1].

Nowadays, the population is densely populated, and the road system is complex. Traffic accidents may occur because of many reasons, such as the carelessness of the drivers, the distraction of pedestrians, or the improper design of the road systems. Therefore, it is necessary to have a program that can issue a warning before a few seconds in collisions between pedestrians and vehicles. The result of the program can detect collisions or help transportation engineers to design the road system properly.

Previous studies have focused on traffic accident prediction but did not investigate the input data as real-time video from a camera [2,3]. The use of the camera has attracted many studies related to surveillance cameras, intelligent traffic system cameras, or moving object recognition [4-7]. Therefore, a camera-based traffic collision warning system could be a solution to reduce traffic accidents. Our paper proposes a warning collision system using the camera as input data.

The warning system has two primary modules. The first module uses image processing to identify and track the movement of objects, including pedestrians and vehicles. The inputs of this first module are the videos from the cameras placed at the surveyed locations. The processed data related to object movements become the inputs of the second component. The operation of the first module for people and vehicles tracking has been presented in our previous article [8].

This paper focuses on the second component that uses machine learning to predict collisions and give warnings.

DOI: 10.24507/icicel.16.01.17

The main contribution of this paper is to propose a traffic collision warning system using data collected by the object tracking module presented in our previous paper. Different models are investigated to find a suitable model. A traffic collision warning system using the found model is also experienced.

The paper is presented as follows. Section 2 discusses the research methodology, the algorithm used in this research, and the traffic collision warning system. Section 3 describes the experimental results. Section 4 gives conclusions and future work.

2. Methodology.

2.1. **Proposed method.** This paper investigates Long Short-Term Memory (LSTM) model to forecast the location of vehicles and pedestrians. The proposed system may predict vehicle collisions with other vehicles or with pedestrians. Figure 1 illustrates the steps to build the second component of the warning system.

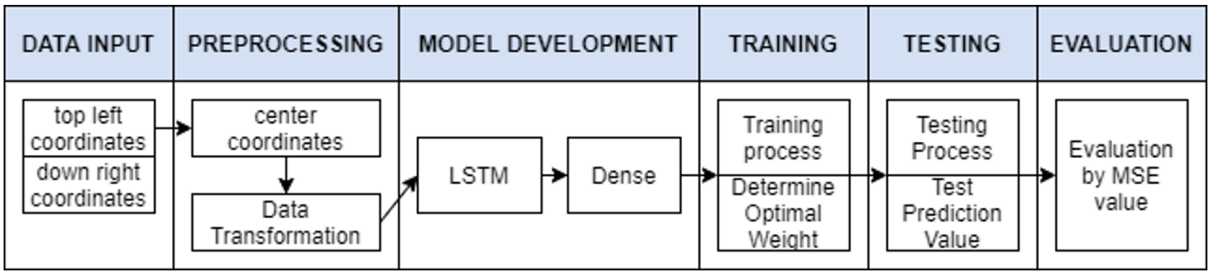

FIGURE 1. Steps to build the position prediction component

The main step was "model development" which investigated three models called One-to-One Vanilla LSTM, Many-to-One Vanilla LSTM, and Neural Network (NN), respectively. After development, the model was used to train and test. To optimize the training process, the parameters were finetuned in our experiments to obtain appropriate parameters for the given dataset. These parameters include epoch numbers and batch sizes.

The model performance evaluation was done by calculating the Mean Square Error (MSE) from the training and testing processes. The MSE was used to measure the consistency of the model with square differences between actual data and predicted data as shown in Equation (1).

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(X_{obs,i} - X_{pred,i})^2 \tag{1}$$

where n is the number of datasets, $X_{obs,i}$ is the observed value, and $X_{pred,i}$ is the predicted values. The lower MSE means the performance of the model is better and the value of the prediction was close to the ground truth value.

Finally, the best performing model in our experiments was used in the collision warning system.

2.2. **Dataset.** The dataset comes from the module that was presented in our previous journal paper [7]. This data consists of top-left coordinates and down-right coordinates of the bounding boxes. There are 745198 samples for training and 319372 samples for testing. Each data sample is a one-dimensional array of N elements ($N = 20, 30, 40$). Each array is the position coordinates of a vehicle or a pedestrian in time series separated by 1/30 second. The example of data that has 20 elements is shown in Figure 2. In this situation, the first 19 elements are used as input and the final element is considered as output data.

The outputs of the detection component are the bounding boxes of vehicles and pedestrians. Each bounding has four elements. The first two elements (x_start, y_start) are

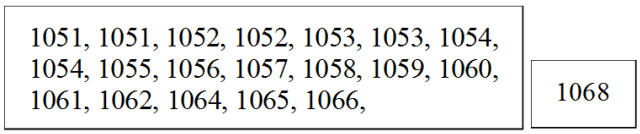

FIGURE 2. Data sample with 20 elements

reserved for the top-left corner coordinates of the bounding box while the two remaining elements (x_end, y_end) are used as the lower-right coordinate of each bounding box.

In the detection phase, when the system cannot detect an object in one frame, four coordinates of the object become "−1". However, the use of all four features (x_start, y_start, x_end, y_end) requires much time and effort. Hence, the number of features is reduced by preprocessing phase which has two tasks.

The first task is to calculate the center point of each bounding box as can be seen in Equations (2) and (3).

$$x\_center = \frac{x\_start + x\_end}{2} \tag{2}$$

$$y\_center = \frac{y\_start + y\_end}{2} \tag{3}$$

The second task deals with the interruption points which has value of "1" that can be seen in Equation (4). During the center point calculating:

- If there is an interruption point, the value of this point will be assigned to the nearest positive integer value in the future.
- If the nearest positive integer value is also "−1", the value of the interruption point will be assigned to the nearest positive value in the past.

$$\begin{aligned} &If\ (x\_center_i = -1)\ Then \\ &\quad if\ (x\_center_{i+1} > -1)\ Then\ (x\_center_i = x\_center_{i+1}) \\ &\quad Else\ (x\_center_i = x\_center_{i-1}) \end{aligned} \tag{4}$$

Figures 3 and 4 present the values of one data sample before and after preprocessing.

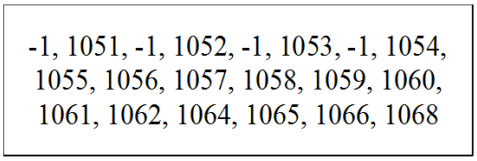

FIGURE 3. Values of one data sample before the preprocessing phase

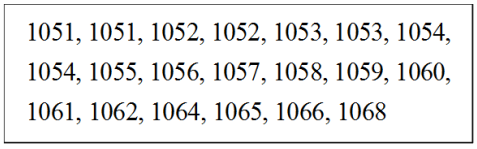

FIGURE 4. Values of one data sample after the preprocessing phase

2.3. **Model development.**

2.3.1. *Neural Network (NN).* Many articles and magazines related to the use of machine learning in various life issues have been published [9-13]. One of the most common aspects of machine learning is the neural network which simulates the activity of the human brain [14,15]. NN has been used in traffic speed prediction [16,17], vehicle trajectory prediction [18], or vehicle reliability [19].

2.3.2. *Long Short-Term Memory model (LSTM).* LSTM is a special recurrent neural network capable of learning long-term dependencies and is currently being used in a variety of domains to solve sequence problems [20]. A Vanilla LSTM is a simple LSTM configuration that consists of an input layer, one fully connected LSTM hidden layer, and one fully connected output layer as can be seen in Figure 5. Vanilla LSTM is the LSTM architecture that may obtain high accuracy on small sequence prediction problems.

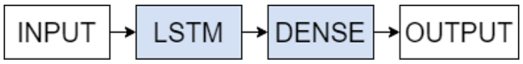

FIGURE 5. Vanilla LSTM architecture

Sequence problems can be broadly categorized into the following categories: One-to-One, Many-to-One, One-to-Many, and Many-to-Many. Each sequence is used to solve different problems. An LSTM network has many LSTM memory cells linked together. In our problem, it is needed to predict one position of vehicles and people in the future so that the One-to-One and Many-to-One are employed.

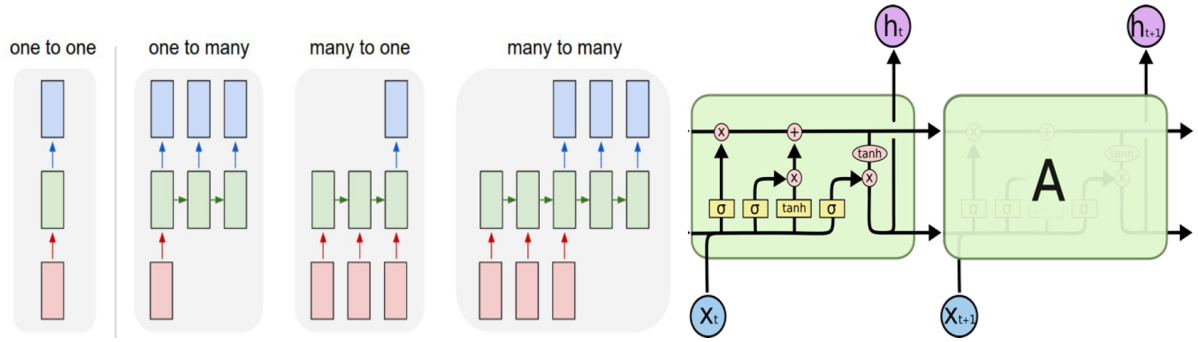

FIGURE 6. Diagram of different sequence types [21] and structure of LSTM [22]

The input of each LSTM layer has samples, time steps, and features. One sample is one sequence, and a batch consisted of one or more samples. One time step is one point of observation in the sample. One feature is one observation at a time step. In this paper, One-to-One Vanilla LSTM is called LSTM01 while Many-to-One Vanilla LSTM is considered as LSTM02.

Figure 7 presents the LSTM01 architecture and Figure 8 shows the LSTM02 architecture. In LSTM01, $\boldsymbol{x_1}$ is the number of memory cells, $\boldsymbol{x_2}$ is the number of features and the number of time steps equals 1. In LSTM02, $\boldsymbol{x_1}$ is the number of memory cells, $\boldsymbol{x_2}$ is the number of time steps and the number of features equals 1. After $\boldsymbol{x_1}$ and $\boldsymbol{x_2}$ have been determined, the model is trained with the number of epochs $\boldsymbol{x_3}$ and batch size $\boldsymbol{x_4}$.

2.4. **Warning collision system.** After selecting a suitable model from experiments, a collision warning system will be developed as shown in Figure 9.

In the architecture, the predicted center coordinates of the bounding boxes are $((X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n))$. The warning system will notify "possible collision" when either of the conditions in Equation (5) or (6) is satisfied.

$$|X_a - X_b| \leq (W_a/2 + W_b/2 + C) \tag{5}$$

$$\text{or} \quad |Y_a - Y_b| \leq (H_a/2 + H_b/2 + C) \qquad (C = 2) \tag{6}$$

where (X_a, Y_a) and (X_b, Y_b) are the coordinates of bounding boxes a and b, (W_a, H_a) and (W_b, H_b) are the width and height of bounding boxes a and b. The error of the formulas, C, is set to 2. The value of "2" is selected because "2" is equivalent to 1% of the maximum possible frame value of 1600×1080.

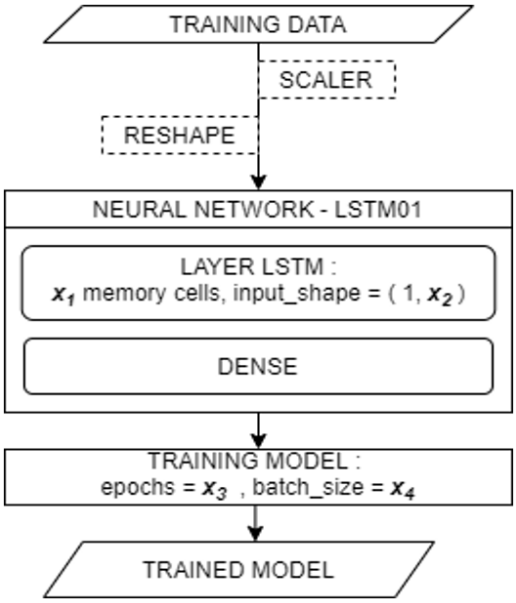

FIGURE 7. LSTM01 architecture

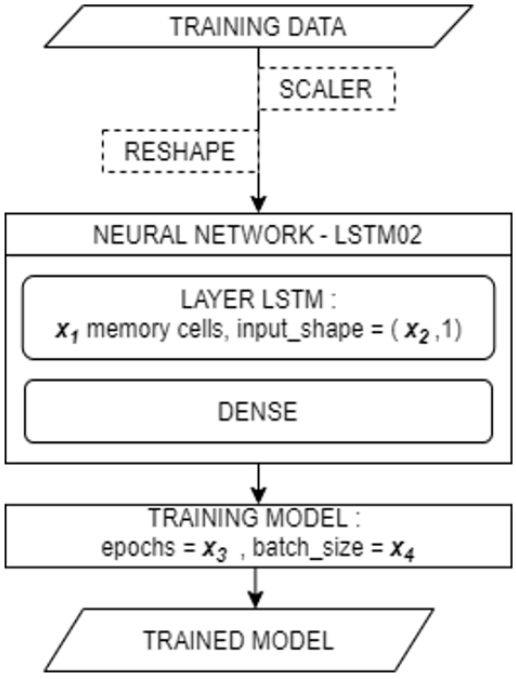

FIGURE 8. LSTM02 architecture

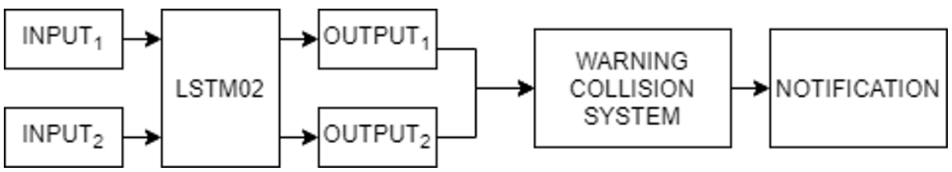

FIGURE 9. Warning collision system

3. **Result and Discussion.** Different models are used in each experiment, and each model is given a unique name. For NN, the use of "x01" means the scaling data step was used in the range from 0 to 1, which is used to distinguish it from NN networks without

scaling data steps in the range from 0 to 1. For LSTM, data scaling is indispensable so that all data used in LSTM01 and LSTM02 were scaled to the range $(0, 1)$.

The first number is the number of input nodes, for example, "20" means the network has 20 inputs. For NN, the second and third numbers indicate the node numbers in each hidden layer, respectively. As for LSTM, the second number is memory cell numbers ($\boldsymbol{x_1}$). The $\boldsymbol{x_2}$ values of LSTM01 or LSTM02 are the input numbers. With LSTM 03, "li" means the linear activation function. Finally, "eps" and "bz" are epoch number and batch size of each model. For example:

- Model 1 (NN_20_60_30_eps6_bz2): an NN model with 20 input nodes, 60 nodes in hidden layer 1, 30 nodes in hidden layer 2, the number of epochs is 6 and the batch size is 2.
- Model 12 (LSTM02_20_10_eps200_bz32): an LSTM02 model with 20 input nodes, 10 memory cells, the number of epochs is 200, and the batch size is 32.

Our experiments investigated two different situations. The first situation used 19 frames as input frames and the 20th frame was used as the predicted frame. On the other hand, 39 frames were considered as input frames and the 40th frame was the predicted frame in the second situation.

Table 1 shows the experimental results. For example, x_center_20 means the predicted data in the x-coordinate of the center point using 19 frames as input data. The results demonstrated that if the model obtained high accuracy with center_test_20, the model also got high accuracy with center_test_40. In addition, model 9 and model 10 in center_test_20 got the lowest errors compared with other models. Compared with model 9, model 10 achieved a little better error in center_test_20 but much worse in center_40. Therefore, model 9 could be the better model for the warning collision prediction system.

TABLE 1. Experimental result (Test model on window 10 – 64 bit, Dell Precision M4800, Intel® Core i7® Processor 4900 QM – 3.8 GHZ, RAM 8G)

No.	Model's name	x_center_test_20	x_center_test_40	y_center_test_20	y_center_test_40
1	NN_20_60_30_eps6_bz2	2.337	29.287	4.875	72.392
2	NN_20_60_30_eps200_bz64	2.284	29.618	4.676	72.224
3	NN_x01_20_60_30_eps20_bz2	2.045	4.823	1.189	2.858
4	NN_x01_20_60_30_eps200_bz32	2.636	24.423	1.675	17.318
5	NN_x01_20_100_50_eps200_bz32	2.964	25.862	1.943	19.085
6	LSTM01_20_40_eps6_bz1	23.361	236.955	13.618	139.435
7	LSTM01_20_40_eps100_bz32	3.031	44.946	2.005	33.392
8	LSTM02_20_1_li_eps6_bz1	2.111	10.409	1.308	7.178
9	LSTM02_20_2_li_eps4_bz1	1.939	4.731	1.123	2.897
10	LSTM02_20_4_li_eps6_bz1	1.911	6.632	1.119	3.890
11	LSTM02_20_10_eps4_bz1	2.168	15.826	1.327	9.882
12	LSTM02_20_10_eps200_bz32	2.305	16.213	1.377	8.831

Not only two LSTM02 models (model 9 and model 10) gained high accuracy in experiments, but model 3 of NN also earned quite good results. The accuracy of model 3 was better than model 10 and only slightly inferior to model 9 in the center_test_40 situation.

Table 1 also shows that all NN models except model 3 had much bigger errors than LSTM models. Therefore, the LSTM model, especially model 9, could be a solution for the model of warning collision prediction system.

The training time was also investigated. This time is proportional to epoch number and inversely proportional to the batch size. Experimental results showed that all models with high accuracy required training time from 7 to 8 hours.

Using model 9 in the proposed warning collision prediction system, we can get the results seen in Figure 10 when we predict two objects called a and b. The numbers in

X_a	Y_a	X_b	Y_b	W_a	H_a	W_b	H_b
1361	553	1410	472	18	76	73	237
1360	553	1409	471	17	77	75	238
1360	553	1407	470	15	76	74	238
1359	553	1406	469	13	76	73	237
1359	553	1405	469	12	76	73	237
1357	553	1402	469	11	76	72	236
1356	553	1401	468	11	76	72	236
1356	553	1400	468	11	76	72	236
1355	553	1397	467	11	76	72	236
warning							
1355	553	1397	467	11	76	72	236
warning							
1353	553	1396	467	11	76	72	236

FIGURE 10. Operation of the warning system

this figure are the position (x, y), width, and height of each object bounding box. The system will notify "warning" if a collision occurs.

4. **Conclusion.** This paper focused on the second component of the traffic collision prediction system. The input data of the second component come from the real-time object tracking proposed by our previous paper. This paper proposes a suitable model for predicting traffic collisions from the input data by investigating three different models called NN, One-to-One LSTM, and Many-to-One LSTM. Experimental results showed that the LSTM02 with the configuration of parameters as model 9 obtained the highest performance. Model 9 was also used in our experiments as the model for the warning system.

A possible avenue for future research will focus on the finetuning of the algorithms to obtain better performance. The results of this study could be applied to warning systems located at power poles, traffic lights, wearable devices, and applications installed in cell phones to support and give a warning for pedestrians. Besides, the results of this study can be used on road inspections. At each test point, the frequency of alerts will be counted. If the number of warnings exceeds a predetermined threshold value, it is necessary to change the road such as adding fences, expanding roads, moving the pause line up or down.

REFERENCES

[1] *https://www.who.int/gho/road_safety/en/*, Accessed in May 2021.

[2] D.-J. Lin, M.-Y. Chen, H.-S. Chiang and P. K. Sharma, Intelligent traffic accident prediction model for Internet of vehicles with deep learning approach, *IEEE Trans. Intelligent Transportation Systems*, DOI: 10.1109/TITS.2021.3074987, 2021.

[3] L. Yu, B. Du, X. Hu, L. Sun, L. Han and W. Lv, Deep spatio-temporal graph convolutional network for traffic accident prediction, *Neurocomputing*, vol.423, pp.135-147, DOI: 10.1016/j.neucom.2020.09.043, 2021.

[4] L. T. Dang, E. W. Cooper and K. Kamei, Development of facial expression recognition for training video customer service representatives, *2014 IEEE International Conference on Fuzzy Systems*, pp.1297-1303, DOI: 10.1109/FUZZ-IEEE.2014.6891864, 2014.

[5] A. F. Ahmed et al., An intelligent traffic system for detecting lane based rule violation, *2019 International Conference on Advances in the Emerging Computing Technologies (AECT)*, 2020.

[6] P. Lu, Y. Ding and C. Wang, Multi-small target detection and tracking based on improved YOLO and SIFT for drones, *International Journal of Innovative Computing, Information and Control*, vol.17, no.1, pp.205-224, 2021.

[7] T. L. Dang, T. Cao and Y. Hoshino, Classification of metal objects using deep neural networks in waste processing line, *International Journal of Innovative Computing, Information and Control*, vol.15, no.5, pp.1901-1912, 2019.

[8] T. L. Dang, G. T. Nguyen and T. Cao, Object tracking using improved deep_sort_yolov3 architecture, *ICIC Express Letters*, vol.14, no.10, pp.961-969, 2020.

[9] M. Bkassiny, Y. Li and S. K. Jayaweera, A survey on machine-learning techniques in cognitive radios, *IEEE Communications Surveys & Tutorials*, vol.15, no.3, pp.1136-1159, DOI: 10.1109/SURV.2012.100412.00017, 2013.

[10] S. Sun, A survey of multi-view machine learning, *Neural Computing and Applications*, vol.23, nos.7-8, pp.2031-203, DOI: 10.1007/s00521-013-1362-6, 2013.

[11] Z. Stefanos, C. Zhang and Z. Zhang, A survey on face detection in the wild: Past, present and future, *Computer Vision and Image Understanding*, vol.138, pp.1-24, DOI: 10.1016/j.cviu.2015.03.015, 2015.

[12] J. Qiu, Q. Wu, G. Ding et al., A survey of machine learning for big data processing, *EURASIP J. Adv. Signal Process.*, DOI: 10.1186/s13634-016-0355-x, 2016.

[13] A. L. Buczak and E. Guven, A survey of data mining and machine learning methods for cyber security intrusion detection, *IEEE Communications Surveys & Tutorials*, vol.18, no.2, pp.1153-1176, DOI: 10.1109/COMST.2015.2494502, 2016.

[14] S. O. Haykin, *Neural Networks and Learning Machines*, 3rd Edition, Upper Saddle River, 2009.

[15] A. Martin and P. L. Bartlett, *Neural Network Learning: Theoretical Foundations*, Cambridge University Press, 2009.

[16] J. Tang, F. Liu, Y. Zou, W. Zhang and Y. Wang, An improved fuzzy neural network for traffic speed prediction considering periodic characteristic, *IEEE Trans. Intelligent Transportation Systems*, vol.18, no.9, pp.2340-2350, DOI: 10.1109/TITS.2016.2643005, 2017.

[17] C. Song, H. Lee, C. Kang, W. Lee, Y. B. Kim and S. W. Cha, Traffic speed prediction under weekday using convolutional neural networks concepts, *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp.1293-1298, DOI: 10.1109/IVS.2017.7995890, 2017.

[18] D. Jeong, M. Baek and S. Lee, Long-term prediction of vehicle trajectory based on a deep neural network, *2017 International Conference on Information and Communication Technology Convergence (ICTC)*, pp.725-727, DOI: 10.1109/ICTC.2017.8190764, 2017.

[19] N. Akail, L. Y. Moralesl and H. Murase, Reliability estimation of vehicle localization result, *2018 IEEE Intelligent Vehicles Symposium (IV)*, pp.740-747, DOI: 10.1109/IVS.2018.8500625, 2018.

[20] R. Quan, L. Zhu, Y. Wu and Y. Yang, Holistic LSTM for pedestrian trajectory prediction, *IEEE Trans. Image Processing*, vol.30, pp.3229-3239, DOI: 10.1109/TIP.2021.3058599, 2021.

[21] R. Miotto, F. Wang, S. Wang, X. Jiang and J. T. Dudley, Deep learning for healthcare: Review, opportunities and challenges, *Briefings in Bioinformatics*, vol.19, no.6, pp.1236-1246, DOI: 10.1093/bib/bbx044, 2018.

[22] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural Computation*, vol.9, no.8, pp.1735-1780, DOI: 10.1162/neco.1997.9.8.1735, 1997.