Motion Prediction for Autonomous Vehicles using Multilayer Perceptron Model and Long Short Term Memory(LSTM)

Aditya Gowda, *M.Sc., Commercial Vehicle Technology, TU Kaiserslautern*
Kushal Bollempalli, *M.Sc., Commercial Vehicle Technology, TU Kaiserslautern*
Rahul Banbe, *M.Sc., Commercial Vehicle Technology, TU Kaiserslautern*

***Abstract*—With developments in technology, automated vehicles are increased in human surroundings, the ability of motion prediction type systems is important to perceive and understand the human behavior during crossing road or walking situations. By keeping in mind, Improvement in road safety automobile sector mainly focuses on intelligent automated vehicles and the major matter is to detect hazardous situations and respond accordingly in order to avoid or reduce the accidents. To detect potentially dangerous situations as early as possible, it is therefore to know the position and motion of the ego vehicle and vehicle surrounded it for several seconds in advance, for this purpose a long-term prediction approach based on combined trajectory classification and particular framework. During traffic situations, we can approximately take an overview of position of vehicle and evaluation of prediction of vehicle and risk involvement also contemplate while predicting the motion of vehicle. Depending on driver intention and current set of motion of vehicle, an infinite set of possible future trajectories are obtained and based on that dataset we can predict the direction of vehicle. As there are many methods such as physics based, maneuver based, interaction based to predict the vehicle movement. The method of directly predicting possible target positions automatically includes the environment, which can improve long-term motion prediction performance over other approaches. The main goal is to reduce the motion prediction error of automated vehicles during traffic and surrounded circumstances.**

***Index Terms*—Motion prediction, Risk, Road Safety, Autonomous vehicle, Prediction error**

I. Introduction

Autonomous driving is based on the intelligent and connected capabilities of current and future vehicles. In autonomous vehicles and Advanced driver assistant systems, safety is the most important factor [2]. One of the major expectations from autonomous vehicles is to reduce number of accidents so for this purpose, it is necessary to predict the movements of surrounding vehicles and people for the vehicle to navigate safely and to perform this check, we need to compute the set of states that can be reached within a finite or infinite time interval [3]. The basic approach of this study is to utilize this motion pattern learning ability to estimate future object positions based on motion history [4]. Autonomous vehicles require a large amount of machine learning capabilities for action plans [5]. The main constraint is to use the map information as state constraints and to incorporate road information directly into the state estimation process using variable structure interacting multiple models' methods or Multi-Hypothesis Tracking (MHT) method [5]. Vehicle speed and road curvature enables us to define input functions to estimate the future trajectories for the dynamic system [5]. Sometimes driver has poor driving practices such as changing lanes without signal, which creates difficulty for motion prediction therefore, Advanced driver assistance systems (ADAS) must be able to accurately predict the future behaviour of multiple road users simultaneously under different driving scenarios. This ensures a safe, comfortable and cooperative driving experience [6]. Enabling autonomous vehicles to navigate dynamically changing traffic scenes requires an intention prediction module that can adapt to different scenarios in different driving manoeuvres [6].
Fig. no.1 visualizes insertion areas under different driving scenarios for predicted vehicle.

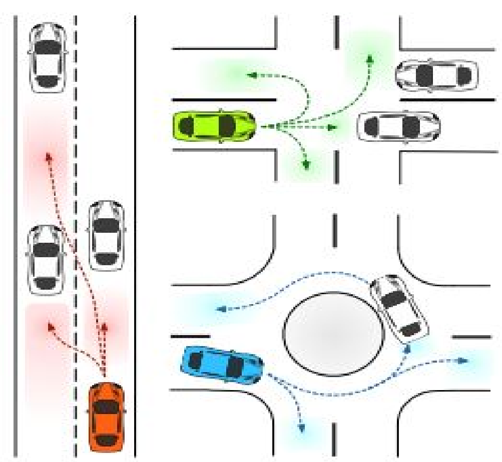

Fig. 1. coloured regions under different environments [6, fig.1]

Motion prediction is primarily treated as a regression problem that seeks to predict the short-term motion and long-term trajectory of a vehicle and by combining motion prediction and intent estimation, it is possible to obtain not only advanced behavioural information, but also the predicted future state of the vehicle [6]. As vehicle operates on a constrained environment of road networks, future trajectories are influenced by road layout and assuming a complaint behaviour of traffic participants, knowledge

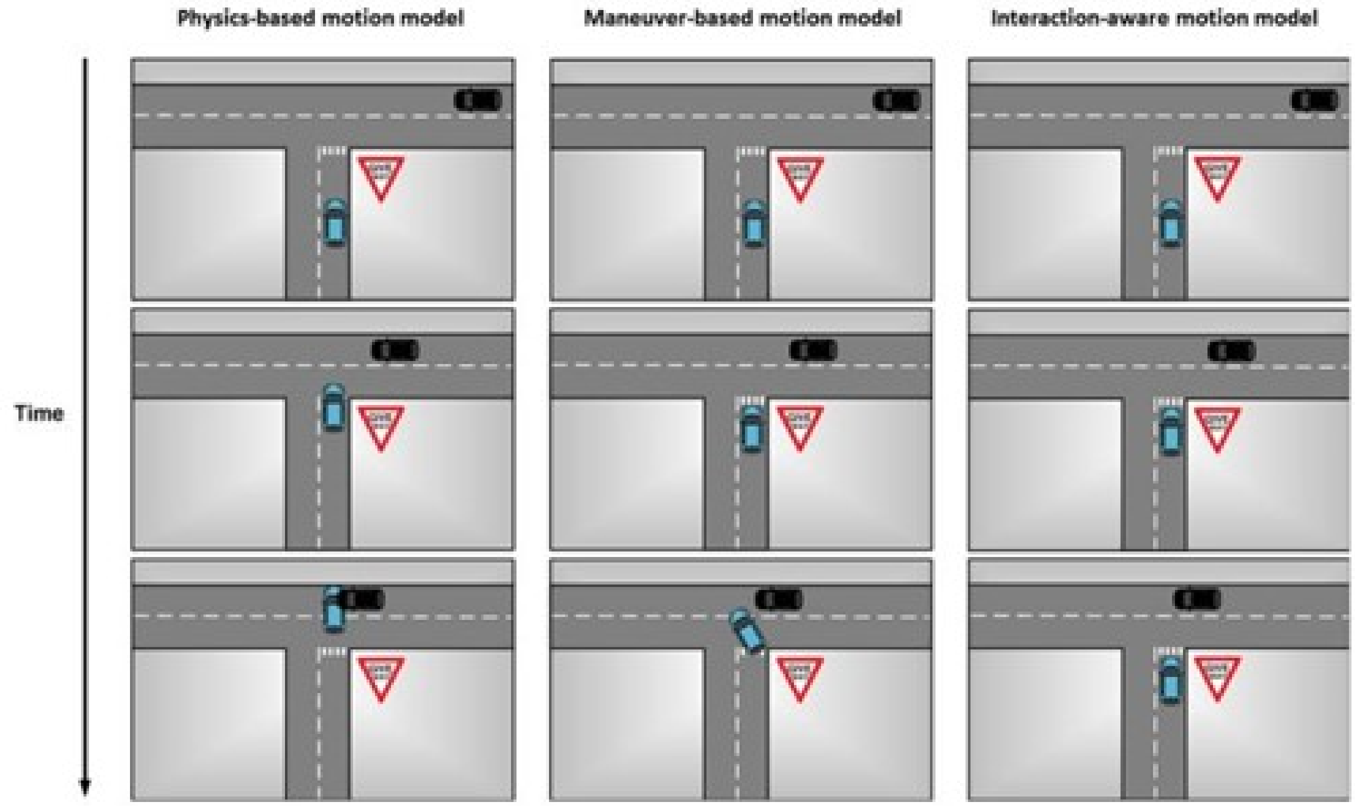

Fig. 2. Motion prediction models [2, fig.2]

of road layout permits long term motion prediction [5]. The Autonomous Vehicle Motion Planning Module uses environmental awareness information provided by multiple sensors to generate a safe, stable, comfortable and feasible reference trajectory for the trajectory tracking module [7]. For autonomous controlled vehicles, motion planning is the key technology. The complexity of finding the optimal path depends on the state-space dimensional, the number of obstacles, and the size of the state-space [7]. The main advantage of suggesting destinations instead of trajectories is that it allows to represent various dynamics and to automatically incorporate environment constraints for unreachable regions [6]. The key to trajectory-based motion prediction is an efficient matching technique for comparing vehicle histories to data sets [4].

We have chosen to organize motion modelling and prediction approaches according to modelled entities are as

• Physics based motion model
• Manoeuvre based motion model
• Interaction aware motion model

Physics-based motion models are the simplest models, they consider that the motion of vehicles only depends on the laws of physics [2]. Typical physics based model includes Constant Velocity model, Constant Acceleration model, constant turn rate and velocity model, constant turn rate and acceleration model [7] . Although physics-based motion models have good real time efficiency, they are insufficient to describe changes in vehicle motion due to abrupt manoeuvring behaviours or environmental factors [7].

Manoeuvre-based motion models are more advanced as they consider that the future motion of a vehicle also depends on the manoeuvre that the driver intends to perform [2]. Typical prediction methods include support vector machine, hidden Markov model and Dynamic Bayesian network however manoeuvre-based models do not consider among traffic participants [7].

Interaction-aware motion models consider the inter-dependencies between vehicles' manoeuvres [2]. an interaction-aware motion model for surrounding vehicles, which modelled the interaction among vehicles with a performance function penalizing the possible collisions [7] .

Physics- and manoeuvre-based models focus on predicting the future motion of individual targets, without considering interactions between neighbouring vehicles. Therefore, a possibility exists of misunderstanding targets' behaviours when multiple vehicles are driving closely together. Interaction-aware models reflect interactions between surrounding vehicles and predict future motions of detected vehicles simultaneously as a scene. These models can predict the most realistic behaviour by reflecting interactions between vehicles.[8]

The main goal of this project is to predict the behaviour of on-road traffic objects such as cars, trucks and bicyclists. The behaviour of traffic objects is constrained by their inertia, driving rules and road geometry and for prediction perception

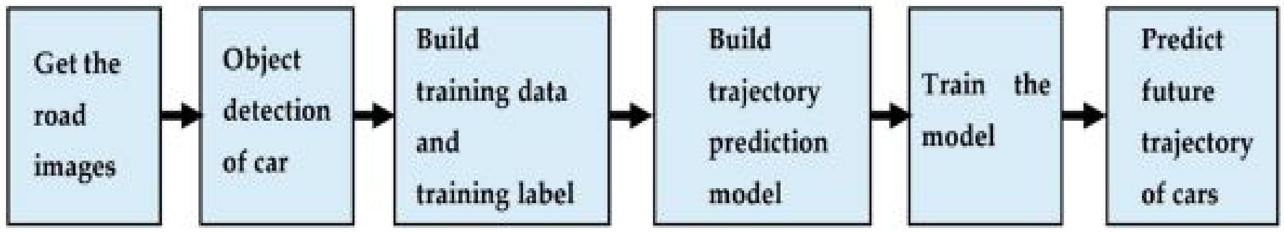

Fig. 3. model of Trajectory prediction of a car [12, fig.16]

of surroundings is crucial.
Block diagram for motion prediction model: The Outline of the motion prediction model is given by figure 3.

The remainder of the paper is organized as follows: Section 2 describes the literature review on prediction models. Section 3 illustrates proposed basic models such as LSTM model, Multilayer Perceptron model and CNN model and section 4 includes steps in data driven prediction models to get results. section 5 describes the conclusion of the paper.

II. Literature Review

Zhang et.al. [7] proposed the method of risk evaluation and motion planning used in autonomous vehicles, using motion prediction of surrounding vehicles. Fusing constant turn rate, acceleration-based models and manoeuvre-based motion prediction models were used in his research.The likelihood of a collision occurrence and the severity of a collision event are then combined to provide a risk indicator for collision risk assessment. Two scenarios are applied to the suggested predictive trajectory planning approach to confirm its viability and effectiveness. [7]

Jeong et.al. [8] developed Long short-term memory (LSTM) to predict the future motion of the vehicle. Based on motion history, the LSTM-RNN based motion predictor was used to forecast erratic behaviours of adjacent cars. 11,662 data samples obtained by on-vehicle sensors on an AV traveling in actual traffic were used to train the proposed network. To standardize each element of the input data and scale back to real-world units using the same encoder parameters, these data were processed with an encoder and a decoder. The time to recognize in-lane targets within the intersection improved significantly over the performance of the base algorithms. [8]

Wang et. al. [10], research proposed a predictive manoeuvre planning and control framework for an autonomously controlled vehicle with uncertainties was presented. This framework includes both discrete manoeuvre planning and motion trajectory planning (disturbances and sensor noise). The manoeuvres are then added to a loosely coupled stochastic MPC that is formulated to simultaneously produce the optimal reference selections and the control input trajectories that minimize an objective function subject to traffic constraints and rules involving other objects that are common in public traffic. Several simulation trials demonstrated that manoeuvre planning enables the autonomous vehicle to more effectively adapt to its surroundings. [10]

III. Proposed Models

A. LSTM-RNN Based model:

LSTM- RNN based model focused on a motion predictor for surrounding vehicles in multi-lane turn intersections. Trajectory level prediction which covered in structured environments that enforce behaviour such as lane changing or lane keeping.
Recurrent Neural Network is a network that works on the present input by taking into consideration the previous output (feedback) and storing in its memory for a short period of time (short-term memory). There are several drawbacks of this model such as it fails to store information for a longer period and there is no finer control over which part of the context needs to be carried forward and how much of the past needs to be ‘forgotten’. During training the model, RNNs exploding and vanishing the gradient which is overcome by long short-term memory network (LSTM). Long-time lags in certain problems are bridged using LSTMs where they also handle noise, distributed representations, and continuous values. LSTMs provide us with a large range of parameters such as learning rates, and input and output biases. Hence, no need for fine adjustments.

Trajectory-level prediction is more important in multi-lane turn intersections because different drivers will have varied driving behaviours as they travel through the intersections, as shown in Fig.4. It shows future trajectories of the surrounding vehicles that travel with the subject vehicle are predicted based on the states of subject and surrounding vehicles during multi lanes.

Fig.4 shows future trajectories of the surrounding vehicles that travel with the subject vehicle are predicted based on the states of subject and surrounding vehicles during multi lanes.

B. Multilayer Perceptron model

A multilayer perceptron model is a fully connected class of feed-forward artificial neural network.The neurons are arranged in layers, the connections are always made from lower

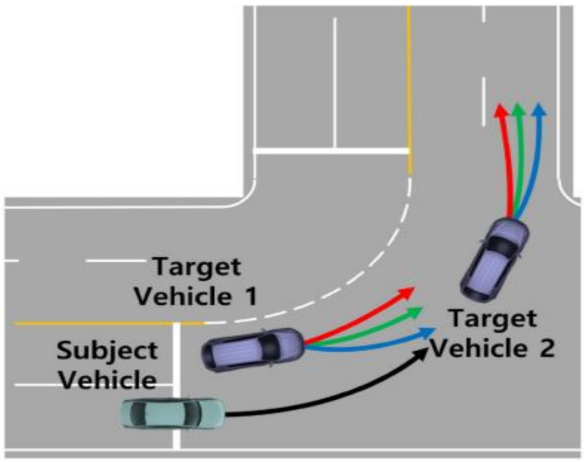

Fig. 4. Motion trajectories of vehicle [8, fig.2]

layers to upper layers, and the neurons within the same layer are not connected and structure has one or more hidden layers between its input and output layers[13].Most of the time, the signals are sent from input to output within the network and there is no loop such as each neuron's output has no impact on the neuron itself. This design is known as feed-forward [14].The term "hidden" refers to layers that are not directly related to their surroundings. Since the input layer's primary job is to transfer input signals to the upper level without any input processing, there is some debate in the reference material on whether it should be considered an independent layer in the network [14]. Feed forward Multilayer model is given by fig.5

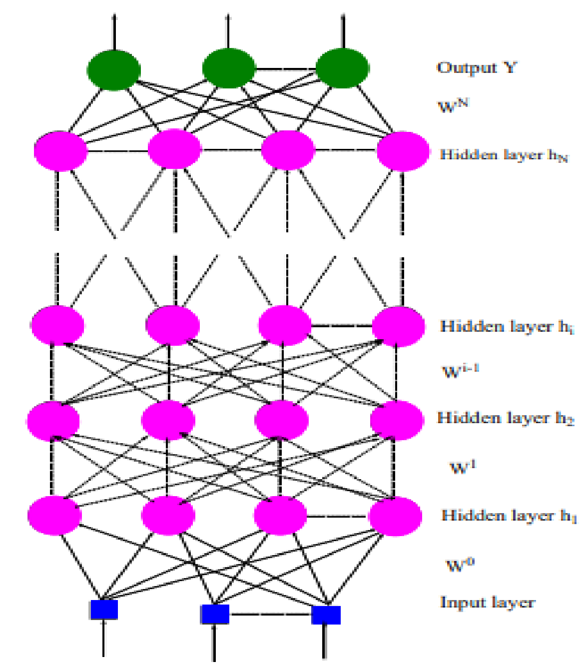

Fig. 5. Feed forward multilayer perceptron model [13, fig.1]

C. CNN Model

By convoluting the kernel (weight filter) on the input image, CNN calculates the feature map corresponding to the kernel. Given that there are several kernels, feature maps corresponding to the various kernel types can be generated. Fully connected layers receive the extracted feature map as an input, and the probability for each class is then output. In this instance, the network topology between the input and output layers comprises units for the picture and the number of classes.[10]

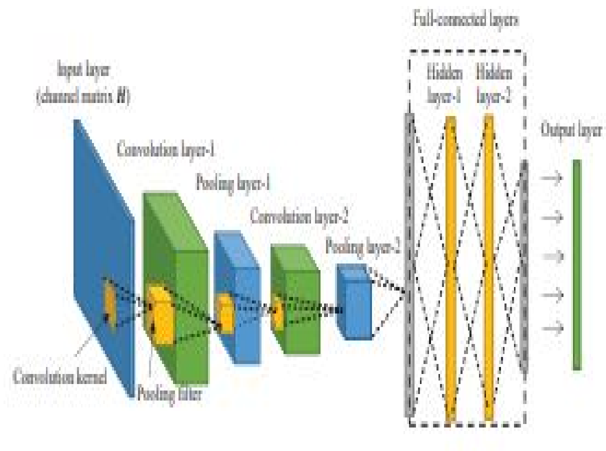

Fig. 6. Basic CNN Structure [11, fig.1]

Fig.6 illustrates the basic structure of CNN which consists of five main components:

Input layer: CNN reorganizes the input data into a 3-dimensional matrix that resembles a picture. The length and breadth of the image are represented by the first two dimensions, while the depth is indicated by the final dimension. The input information for P1 and P2 refers to the channel matrix H below. The channel coefficients can be further standardized and transformed to dB to simplify the training process.[11]

Convolution layer: Each neuron in the convolution layer only connects to a squared portion of the layer above. The filter or convolution kernel is referred to as this squared core. The number of parameters in the neural network can be greatly reduced because the weights are shared via the convolution kernel. [11]

Pooling layer: The output matrix from the preceding convolution layer is further reduced in size using the pooling layer. A filter is used by the pooling layer, just like the convolution layer, to transform a node matrix into a unit node. Instead of applying convoluting processes, the pooling filter uses maximizing or averaging operations. [11]

Fully connected layer: Convoluting and pooling can be thought of as an automatic feature extraction procedure. Then, in order to produce the output, full connected (FC) layers are also required. FC layer's structure is the same as DNN's. [11]

Output layer: The output must include estimated data in order to make the best grouping judgments. [11]

IV. STEPS IN DATA DRIVEN PREDICTION MODELS

In this part, we will outline the steps we took to use the LSTM model and the multi-Perceptron model to determine the best outcome. The suggested method for deep learning-based autonomous vehicle motion planning's comprehensive network architecture. We enter the multi-frame image segment into the system for network processing, and after traversing the spatiotemporal LSTM network, we eventually obtain the average displacements error, average final displacement, and average absolute heading error. We need to execute pre-processing before entering the system since the initial information we got is single-frame image data that the camera gathered. To create a multi-frame image segment based on time series, single-frame picture data is integrated in time series during the pre-processing stage. The system then receives the multi-frame picture segment based on a time series, followed by 4 layers of LSTM. The output of the motion planning result is finally calculated using a single layer of CNN.
In the following 7 parts, we got the best result:
1. Importing Libraries, Local Libraries, and Data.
For code reuse and the solution for these modules, we must divide the program into many files as its size increases. So, in order to handle the data, we imported all the required modules. The imported modules are padas, numpy, xlsxwriter, os, and sys. Following that, we add the data processing, the necessary files, and the dataset location.

2. Reading Data
To determine which recording id produces the least amount of error, we went through a lot of iterations. We imported the recordings with IDs 26, 27, and 29 since their displacement and angle errors were the least.

3. Pre-processing and Downsampling the Data
It sometimes arises while working with data that we receive unusual, loud, unwanted, huge, unstructured Excel sheet portions. The task is to utilize the chosen id location data and transform it into knowledge. The dataset is presenion tted in a specific sequence, but in order to deal with it, we must organize it well. In order to extract some information beforehand, we build a "pre_$process_obj$" and give it the data. The system device cannot handle every frame of data; thus, we utilized every k^{th} piece of data while preserving all the crucial information. Given the capability of the system's computing power, we have skipped 5 frames when going through the dataset's size.

4. Preparing Data for Training and Testing
Our prediction algorithm should be fed some historical data from x and forecast the result y. We can determine which y should be projected based on each x using the dataset. First, we load and provide the data object with all the required information. Regarding "overfitting," it is crucial to divide the data into separate portions. Thus, the "data stacking" and "data prepare obj" functions in this case retrieve the vectors from our time series. Additionally, it creates a single vector by stacking every sequence that was acquired beneath one another. It separates the final, enormous vector into a collection of "training" data and "validation" data as well. So, we allocated a total of 75 track ids. The Pickle library will be used to save and load test data.

5.Prediction models
The new neural network prediction model has four Conv-LSTM layers with neurons that are 52, 42, 34, and 10 correspondingly. The epoch is set at 20, the batch size is 32, and the activation method is 'relu'.

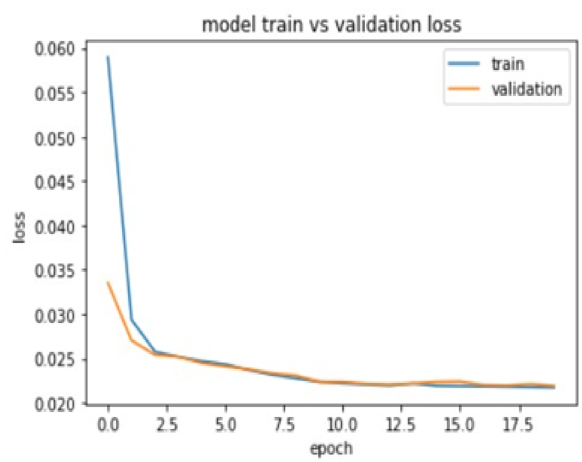

Fig. 7. Model train vs validation loss: LSTM

6.Predcition testing:
We must use our model to generate predictions in order to assess how well we did. We must make forecasts and contrast them with the actual results. The recording '28' was used for the data-driven forecast. Preprocessing and downsampling data are done to match the training dataset, and we also have to add evaluation from our local file data to obtain the ground truth data preparation. This is because we trained on recording location id locations "26," "27," and "29" and tested on recording id location "28." In order to evaluate the model, an excel sheet and prediction function are created.

7. Evaluation
Now that we have loaded the data from the excel sheet, we will compute three error measures; Average Displacement Error (ADE), Final displacement error (FDE), and Average Absolute Heading Error (AHE). By utilizing the correct formulae and inputting the excel sheet, we determine the error:

$$ADE = \frac{\Sigma_{i=1}^{n} \Sigma_{t=T_{Frame}}^{T_{pred}} [(\hat{x}_i^t - x_i^t)^2 + (\hat{y}_i^t - y_i^t)^2]}{n(T_{pred} - (T_{frame} + 1))} \quad (1)$$

$$FDE = \frac{\Sigma_{i=1}^{n}\sqrt{(\hat{x}_i^{T_{pred}} - x_i^{T_{pred}})^2 + (\hat{y}_i^{T_{pred}} - y_i^{T_{pred}})^2}}{n} \tag{2}$$

$$AHE = \frac{\Sigma_{i=1}^{n}\Sigma_{t=T_{Frame}}^{T_{pred}}|\hat{y}_i^t - y_i^t|}{n(T_{pred} - (T_{frame} + 1))} \tag{3}$$

Finally, through evaluation we get,
The average displacement error is 2.715 m
The average final displacement error is 2.629 m
The average absolute heading error is 3.52 degrees.

A. Results and Evaluation

Likewise procedure was followed and we performed motion prediction for the autonomous vehicle using the Multi-perceptron model to evaluate the average displacements error, average final displacement, and average absolute heading error. In order to create a multi-frame image segment based on time series, single-frame picture data is integrated into time series during the pre processing stage. The system then receives the multi-frame picture segment based on a time series, followed by 8 hidden layers of a multi-layer Perceptron model. The output of the motion planning result is finally calculated using a single dense layer.
We imported the same recordings as used in LSTM 26, 27, and 29 since their displacement and angle errors were the least and also obtained similar errors when testing location id 28. The Multi-Perceptron model has 8 hidden layers with neurons that are 900, 750, 550, 375, 250, 100, 50, and 10 correspondingly. The epoch is set at 135 and the activation method for the hidden layer is ‘relu’ while the activation method for the output layer is ‘softmax’. In order to evaluate the model, an excel sheet and prediction function is created.

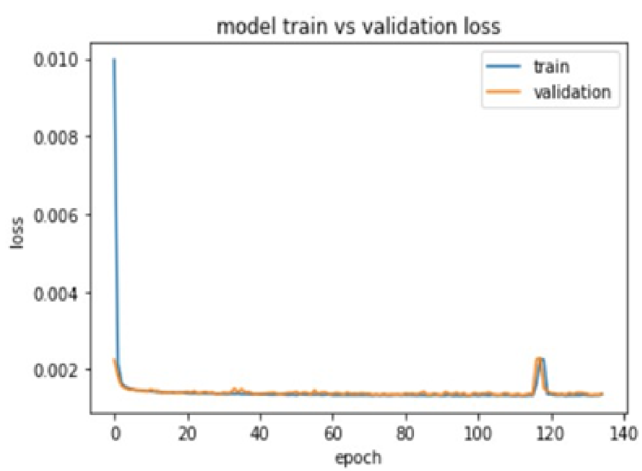

Fig. 8. Model train vs validation loss: FCN

The error result evaluated are as follow:
The average displacement error is 2.717 m.
The average final displacement error is 2.729 m.
The average absolute heading error is 3.48 degrees

V. Conclusion

For Motion Prediction, we have implemented two neural network models i.e., Multilayer Perceptron Model and the LSTM model. For the Multilayer Perceptron model, the resulted errors are; the average displacement error, the average final displacement error, and the average absolute heading error are 2.717 m, 2.729 m, and 3.48 degrees respectively. For the LSTM model, the resulted errors are; The average displacement error, the average final displacement error, and the average absolute heading error are 2.715m, 2.629m, and 3.52 degrees. As both the neural network models have almost the same result LSTM is better as the LSTM cell adds long-term memory in an even more performant way because it allows even more parameters to be learned and also the time for computation is comparatively less.

References

[1] A. Rudenko,L. Palmieri,M. Herman, K. Kitani, D. Gavrila, and K. Arras, ”Human motion trajectory prediction: A survey”, *The International Journal of Robotics Research*, Vol.39, no.8, pp.895-935, 2020.

[2] Lefèvre, Stéphanie and Vasquez, Dizan and Laugier, Christian, Lefèvre,”A survey on motion prediction and risk assessment for intelligent vehicles”, *ROBOMECH journal*, vol.1 , no.1, pp.1-14, 2014.

[3] Medina Lee, Juan Felipe and Trentin, Vinicius and Villagra, Jorge,”Framework for motion prediction of vehicles in a simulation environment”, *XL Jornadas de Automática*, pp.520-527, 2019.

[4] Hermes, Christoph and Wohler, Christian and Schenk, Konrad and Kummert, Franz,”Long-term vehicle motion prediction”, *2009 IEEE intelligent vehicles symposium*, pp.652-657, 2009.

[5] Petrich, Dominik and Dang, Thao and Kasper, Dietmar and Breuel, Gabi and Stiller, Christoph,”Map-based long term motion prediction for vehicles in traffic environments”, *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pp.2166-2172,2013.

[6] Hu, Yeping and Zhan, Wei and Tomizuka, Masayoshi,”Probabilistic prediction of vehicle semantic intention and motion”, *Probabilistic prediction of vehicle semantic intention and motion*, pp.307-313, 2018.

[7] Zhang, Lijun and Xiao, Wei and Zhang, Zhuang and Meng, Dejian,”Surrounding vehicles motion prediction for risk assessment and motion planning of autonomous vehicle in highway scenarios”,*IEEE Access*, vol.8, pp.209356-209376,2020.

[8] Jeong, Yonghwan and Kim, Seonwook and Yi, Kyongsu,”Surround vehicle motion prediction using LSTM-RNN for motion planning of autonomous vehicles at multi-lane turn intersections”, *Surround vehicle motion prediction using LSTM-RNN for motion planning of autonomous vehicles at multi-lane turn intersections*, vol.1, pp.2-14, 2020.

[9] Bai, Zhengwei and Cai, Baigen and ShangGuan, Wei and Chai, Linguo, ”Deep Learning Based Motion Planning For Autonomous Vehicle Using Spatiotemporal LSTM Network” *2018 Chinese Automation Congress (CAC)*, pp.1610-1614, 2018.

[10] Fujiyoshi, Hironobu and Hirakawa, Tsubasa and Yamashita, Takayoshi,”Deep learning-based image recognition for autonomous driving”, *IATSS research*, vol.43 ,no.4 , pp.244-252,2019.

[11] Lei, Lei and Yuan, Yaxiong and Vu, Thang X and Chatzinotas, Symeon and Ottersten, Björn, ”Learning-based resource allocation: Efficient content delivery enabled by convolutional neural network”, *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp.1-5, 2019.

[12] Shen, Chih-Hsiung and Hsu, Ting-Jui,”Research on Vehicle Trajectory Prediction and Warning Based on Mixed Neural Networks”,*Applied Sciences*,vol.11,no.1,pp.7,2020.

[13] R. Hassan,Y. Ghanou, M. Ettaouil, I. Janati and A. Mohammed, ”Multilayer perceptron: Architecture optimization and training”, *International Journal of Interactive Multimedia and Artificial Intelligence*,2016.

[14] M. Popescu, V. Balas,L.Popescu and N. Mastorakis” Multilayer perceptron and neural networks”,*WSEAS Transactions on Circuits and Systems*, vol.8,no.7, pp.579-588, 2009.