International Science Journal of Engineering & Agriculture
2024; 3(4): 38-50
https://isg-journal.com/isjea
doi: 10.46299/j.isjea.20240304.04
ISSN: 2720-6319

Automobile system for predicting the trajectory of surrounding vehicles

Khrystyna Vysotska
Department of Computerized Automatic Systems, Lviv Polytechnic National University, Lviv, Ukraine
ORCID 0009-0002-8567-9747

Adrian Nakonechnyi
Department of Computerized Automatic Systems, Lviv Polytechnic National University, Lviv, Ukraine
ORCID 0000-0002-1873-6337

To cite this article:
Vysotska Khrystyna, Nakonechnyi Adrian. Automobile system for predicting the trajectory of surrounding vehicles. International Science Journal of Engineering & Agriculture. Vol. 3, No. 4, 2024, pp. 38-50. doi: 10.46299/j.isjea.20240304.04.
Received: 07 02, 2024; **Accepted:** 07 31, 2024; **Published:** 08 01, 2024

Abstract: The improvement of autonomous driving systems and advanced driver assistance systems (ADAS) heavily relies on accurate vehicle trajectory prediction. This research focuses on developing a robust method for predicting the trajectories of surrounding vehicles by leveraging machine learning techniques and neural networks. The study integrates data from onboard sensors, vehicle-to-vehicle (V2V) communication, cameras, LiDAR, and Differential GPS (DGPS) to enhance the accuracy and reliability of trajectory forecasts. Traditional approaches, primarily based on physical models, fall short in complex driving scenarios due to their dependency on uniform motion parameters. The investigated approach addresses these limitations by employing separate algorithms to predict longitudinal and lateral positions, thereby improving safety and reducing collision risks. The implemented algorithms were initially tested on simulated data, confirming their functionality. Future steps involve collecting and preparing real-world data to evaluate the algorithms under diverse and complex road conditions. This paper lays the groundwork for future developments in collision avoidance systems and highlights the potential benefits of integrating advanced perception systems in enhancing environmental perception and data quality. The proposed method shows promise in mitigating traffic accidents and optimizing traffic flow, underscoring its importance for future automotive innovations.

Keywords: advanced driver assistance system, vehicle-to-vehicle communication, trajectory prediction, machine learning, onboard sensors, random forest, recurrent neural network.

1. Introduction

Trajectory prediction for vehicles is a critical task in developing autonomous driving systems and advanced driver assistance systems. Considering that the number of cars on the roads constantly increases, modern automobile systems aim to provide higher levels of safety and traffic efficiency. One of the main challenges remains reducing traffic accidents, which can be achieved by improving systems for predicting the behavior of surrounding vehicles. With the help of machine learning technologies and vehicle-to-vehicle communication, there is potential to significantly enhance the accuracy and reliability of trajectory prediction, making this topic extremely relevant for research.

2. Object and subject of research

The object of the research is the prediction of vehicle trajectories. The study focuses on the combined application of various machine learning algorithms and neural networks to enhance the accuracy and reliability of trajectory forecasts. Data from the onboard sensors and vehicle-to-vehicle (V2V) communication messages is utilized to create a robust model capable of accurately predicting future vehicle positions based on real-time inputs. The approach leverages the strengths of different machine learning techniques, including supervised learning for model training and reinforcement learning for continuous improvement.

In addition to the prediction system, the research investigates a comprehensive perception system. The perception framework integrates information from multiple sources, including cameras, LiDAR, V2V communication, and Differential GPS (DGPS), to improve environmental perception accuracy and provide high-quality data for the prediction step. By combining these technologies, the system aims to create a more detailed and precise representation of the vehicle's surroundings.

Despite its potential, the current system has several shortcomings. One notable challenge is the need for large volumes of high-quality data to train the models effectively. Additionally, there are limitations related to the integration of data from diverse sources, which can introduce inconsistencies and affect prediction accuracy. Moreover, the real-time processing requirements pose significant computational demands, which can be a constraint in practical implementations.

3. Target of research

The primary goal of the study is to develop and implement an advanced method for predicting the trajectories of surrounding vehicles. To achieve this objective, the research focuses on several key tasks. The first task is to integrate machine learning techniques and neural networks to enhance the accuracy of trajectory forecasts. By utilizing data from various sources, including vehicle-to-vehicle (V2V) communication and onboard sensors, the method predicts both the longitudinal and lateral positions of vehicles with greater precision.

Another essential task involves improving the data fusion process to ensure consistency and reliability of the input from diverse sources. This includes refining algorithms for real-time data processing to meet the computational demands of practical applications. Additionally, strategies for continuous learning and adaptation are being developed, enabling the prediction system to improve over time with new information.

The ultimate objective is to enhance road safety by providing more precise and reliable trajectory predictions, thereby reducing the risk of collisions. By addressing the identified shortcomings in existing systems, the research aims to contribute to the development of more robust and effective autonomous driving and advanced driver assistance systems.

4. Literature analysis

Autonomous vehicles have seen rapid development over the past decade, particularly in terms of safety and efficiency [1]. Original equipment manufacturers are dedicated to advancing driver assistance systems (ADAS) to reduce traffic accidents. Vehicles equipped with ADAS, such as adaptive cruise control, lane-keeping assistance, and emergency braking systems, are already available. The collision avoidance system (CAS) is one of the most crucial driver assistance systems, capable of predicting collision scenarios and alerting the driver in advance. Development vectors for CAS include forward collision warning, blind-spot monitoring, lane departure warning, cross traffic alert, and pedestrian detection, as depicted in Figure 1.

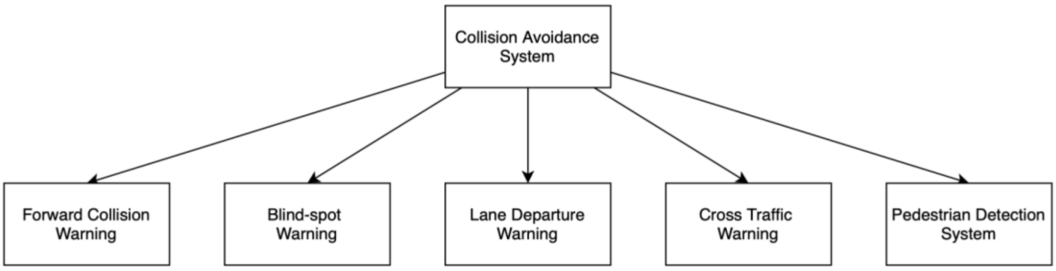

Figure 1. Classification of collision avoidance systems.

A critical task for collision warning systems involves the accurate perception and analysis of the road environment, specifically predicting the trajectories of surrounding vehicles. Trajectory prediction is complex and multifaceted, influenced by driver behavior and specific road conditions. Numerous approaches have been proposed to address this challenge [2].

Traditional methods for trajectory prediction rely on physical models. Dynamic physical models describe movement based on internal vehicle parameters, such as longitudinal and lateral tire forces. Kinematic physical models, used more frequently, describe the vehicle's trajectory based on motion parameters like speed, acceleration, and position. However, kinematic models demonstrate high accuracy only in uniform driving environments, for instance, when surrounding vehicles move with constant speed and acceleration within a single lane. In more complex scenarios, the accuracy of trajectory prediction is significantly reduced.

Machine learning algorithms, particularly recurrent neural networks (RNN), are actively employed for trajectory prediction due to their effectiveness in handling time-series data [3-7]. Many studies utilize the publicly available NGSIM dataset to train prediction models [3-4]. However, the NGSIM dataset contains significant noise, as it was created by processing images from road-mounted cameras [8]. Other studies use the relative coordinates of neighboring vehicles as the output of the prediction model [5-7].

The accuracy of determining the lateral position is a crucial criterion for selecting a trajectory prediction model for collision prevention systems. Even if the prediction model's accuracy has an error of less than one meter, the collision prevention system may fail in a collision scenario. Figure 2 illustrates a situation where the longitudinal position of a neighboring vehicle is predicted correctly, but an error in predicting the lateral position prevents the system from avoiding the collision.

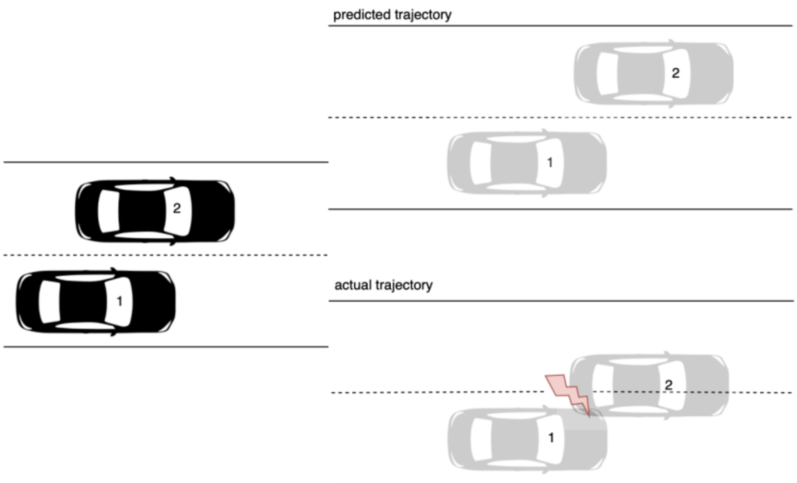

Figure 2. A collision scenario may occur as a result of errors in predicting the lateral position.

This article presents a method for trajectory prediction using separate algorithms to determine longitudinal and lateral positions. The input data includes coordinates of the bounding boxes of surrounding vehicles, obtained from camera sensors, as well as dynamic information received through V2V communication. This information enhances the accuracy of trajectory predictions in complex driving conditions, contributing to the advancement of driver assistance systems and autonomous vehicles.

5. Research methods

The study investigates a system for determining the trajectory of surrounding vehicles using onboard sensor data and V2V messages. The system assumes that vehicles are equipped with V2V communication capabilities, cameras and optionally LiDAR sensors for recognizing the surrounding environment. The proposed structure and data flow are illustrated in Figure 3.

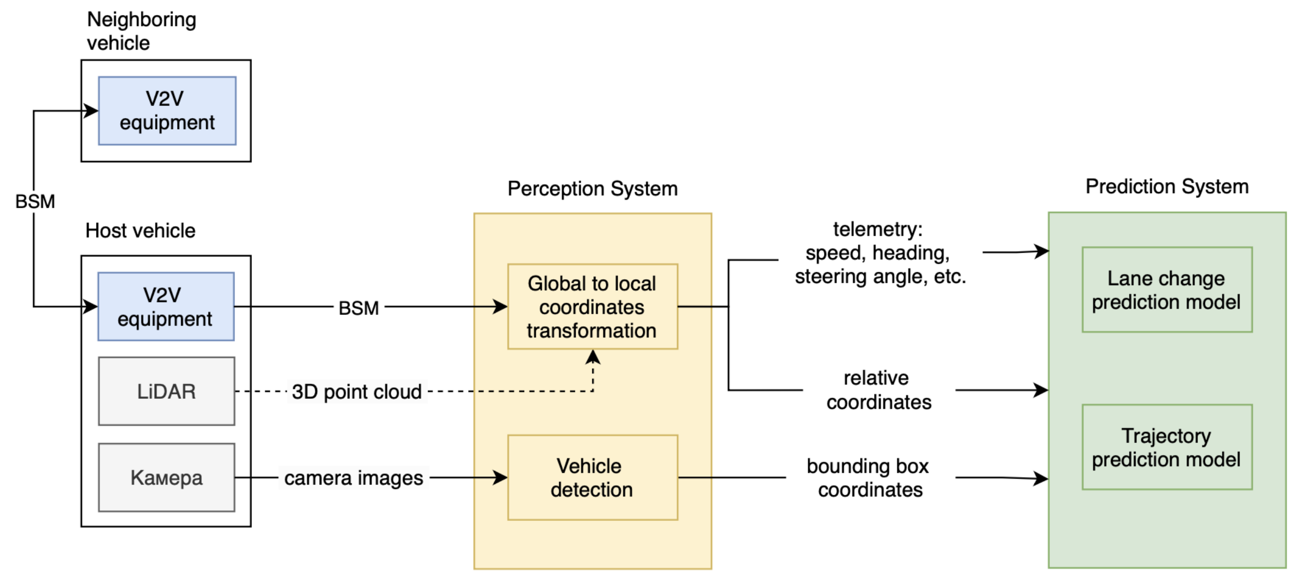

Figure 3. System for determining the trajectory of surrounding vehicles.

The perception system is designed for accurate determination of the position of surrounding vehicles. V2V messages include location data obtained via the Global Positioning System (GPS). However, GPS has vulnerabilities related to environmental factors. For example, when the road is in open areas, the accuracy of location information is about 1 meter, but in urban areas, the location error increases to 5–10 meters due to signal blockage and multipath effects [9].

To address this issue, the use of Differential GPS (DGPS) is proposed, which offers higher accuracy compared to the nominal accuracy of the system. The differential mode involves using two receivers: one is stationary at a known location, referred to as the “base” receiver, while the other remains mobile. Data collected by the base receiver is used to correct the information gathered by the mobile unit. Typically, the base receiver is a professional-grade unit owned by a company licensed to provide navigation services. DGPS can improve location accuracy by receiving correction information from a base station. Since the fixed base station knows its exact location, it can periodically calculate GPS errors and broadcast them.

GPS uses the WGS84 geodetic coordinate system (ECEF-g), which consists of latitude, longitude, and altitude. This system can be transformed into a local tangent plane and applied to a local coordinate system by rotating according to the vehicle's heading angle.

Another method to further improve the accuracy of the perception system is by utilizing LiDAR data. During the process of transforming global coordinates to a local coordinate system, errors can occur depending on the vehicle's heading accuracy. LiDAR sensors generate a 3D point cloud of surrounding objects, providing positional accuracy within less than 10 centimeters. Therefore, the perception system will calculate a more precise position of surrounding vehicles by correlating the location information obtained through V2V communication with the LiDAR data.

To predict the position of surrounding vehicles, a combination of two models is considered: lane change prediction and trajectory prediction. For the lane change prediction model, the use of the Random Forest algorithm is investigated, while for the trajectory prediction model, a Long Short-Term Memory (LSTM) neural network is proposed. The accuracy of predicting the lateral position is crucial for trajectory prediction in a collision prevention system. Even if the predicted relative longitudinal distance is close, it remains a safe state if the vehicle is in an adjacent lane.

Perception System.

In the field of autonomous vehicles, cameras are the most widely used sensors. The primary role of cameras is to detect and classify surrounding objects. A well-known YOLO algorithm can detect objects in real time, since localization and classification are processed simultaneously [10]. The recognition speed of YOLO exceeds 45 frames per second, making it suitable for real-time recognition systems. Considering this, the proposed system also utilizes the YOLO algorithm for environmental recognition and obtaining bounding box values from the camera.

The coordinate values and size of the bounding box correlate with the relative coordinates of the surrounding vehicle, as shown in Figure 4. If the surrounding vehicle is far from the host vehicle, the bounding box size is small, otherwise it is large. Additionally, the position of the bounding box also correlates with the lane in which the vehicle is moving. For example, if the surrounding vehicle is in the adjacent left lane, the bounding box is shifted to the left of the central point of the image. Therefore, in this research, the coordinates and size of the bounding box obtained from the camera are used as input values for the prediction model.

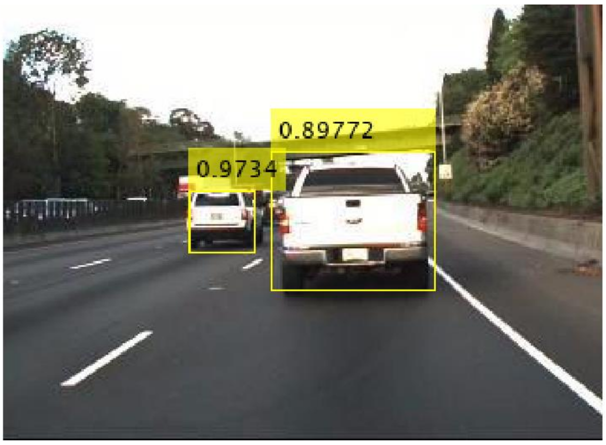

Figure 4. Use of bounding boxes around recognized vehicles at different distances and lanes.

V2V communication represents an important and promising development direction for modern advanced driver assistance systems. V2V communication adheres to the Wireless Access in Vehicle Environment (WAVE) standard defined by IEEE, which is based on IEEE 802.11P and IEEE 1609 standards [11]. With V2V communication using the WAVE standard, vehicles exchange information every 100 milliseconds using the 5.9 GHz DSRC (Dedicated Short Range Communications) frequency band.

The information exchanged between vehicles corresponds to the Basic Safety Message (BSM) — part of the message set j2735 defined by the Society of Automotive Engineers (SAE) [12]. BSM contains dynamic information about the vehicle, such as steering angle and acceleration, as shown in Table 1, which significantly helps to model and analyze the road situation even beyond the visual range of other vehicles. This capability allows for the prediction and prevention of potential collisions, complementing and refining the information from camera and LiDAR sensors in real time, and transmitting critical information for the quickest possible response by drivers.

Table 1. Basic Safety Message (BSM) structure

Part I	Part II
Message Count	Additional details: vehicle events, path history, etc
Temporary ID	Information used by special vehicles such as police cars
Time	Vehicle type, weather information, road hazard information, etc
Position	
Transmission	
Speed	
Heading	
Angle	
Accelset	
Brakes	
Size	

Relative vehicle coordinates can be calculated using GPS information contained in the BSM. The GPS coordinate system is known as the geodetic coordinate system. To transform this system into a local coordinate system with the coordinates of the host vehicle as the origin, it is necessary to convert it to a geocentric coordinate system and then to a local tangent plane. Geocentric coordinates can be derived from the geodetic coordinate system using equations (1)–(3). There are also existing software tools for converting coordinates from the geodetic system to the geocentric system, and vice versa, such as the *geodetic2ecef* function in the Matlab package.

$$x = (N + h)\cos\lambda\cos\varphi\ , \quad (1)$$

$$y = (N + h)\cos\lambda\sin\varphi\ , \quad (2)$$

$$z = (\frac{b^2}{a^2}N + h)\sin\lambda\ , \quad (3)$$

where λ – latitude;

φ – longitude;

h – altitude;

a, b – major and minor semi-axes of the Earth, respectively, according to the WGS-84 standard;

N – normal distance from the Earth surface to the minor semi-axis determined by the equation (4).

$$N = \frac{a^2}{\sqrt{a^2\cos^2(\lambda) + b^2\sin^2(\lambda)}}. \quad (4)$$

The geocentric coordinate system can be converted into the local tangent plane coordinate system using a transformation matrix. Equation (5) calculates the coordinates of the surrounding vehicle relative to the host vehicle.

$$\begin{bmatrix} -sin\,\varphi & -cos\,\varphi & 0 \\ -cos\,\varphi\, sin\,\lambda & -sin\,\lambda\, sin\,\varphi & -cos\,\lambda \\ cos\,\lambda\, cos\,\varphi & cos\,\lambda\, sin\,\varphi & sin\,\lambda \end{bmatrix} \cdot \begin{bmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{bmatrix}, \quad (5)$$

where (x, y, z) – geocentric coordinates of the surrounding vehicle;
(x_0, y_0, z_0) – geocentric coordinates of the host vehicle.

The Matlab package includes the function *ecef2enu*, which implements the described equations and converts geocentric coordinates (x, y, z) into local coordinates relative to a given local point specified by geodetic coordinates.

The transformation of global coordinates into the local coordinate system relative to the host vehicle is completed by rotating the transformation matrix according to the heading angle of the host vehicle. However, if there is an error in determining the heading angle, it results in an error in determining the relative position. To address this issue, the position of the neighboring vehicle can be further assessed using a point cloud obtained through LiDAR.

The point cloud represents a set of points in the XYZ coordinate system around the position of the LiDAR, scanned using multiple rotating laser beams. The accuracy of distance measurement to an object in this way is high, up to 10 cm; however, this requires an additional operation for object classification. Therefore, existing methods for accurate perception of the surrounding environment using LiDAR are mainly divided into three categories: object clustering, object classification, and motion tracking [13].

When two objects are close to each other, there is a probability of recognizing them as one during the object clustering process, as the traditional clustering method classifies objects based on Euclidean distance. Recent studies propose object classification using machine learning algorithms that recognize the shapes and characteristics of objects [14]. This approach provides better results than the traditional clustering method; however, it requires high computational resources and vertical resolution, meaning autonomous vehicles need to be equipped with expensive LiDAR.

To overcome the described limitations, information obtained via V2V communication can be combined with point clouds from LiDAR. Although the calculation of the relative position obtained through V2V communication and GPS may lack precision, the position of the surrounding vehicle can be refined. If clustering is performed around the pre-estimated position, it is possible to distinguish reflected points. Reflected points mostly come from the rear part of the vehicle. When the vehicle changes lanes or drives on a winding road, points reflected from the sides appear. The closest points among the reflections are recognized as the rear part, and the final position can be calculated by taking into account the vehicle's coordinates obtained through V2V communication.

Prediction System.

Vehicle trajectory is represented as time series data which makes recurrent neural network structures widely used in many studies for trajectory prediction [6, 7]. It is necessary to predict the location of the neighboring vehicle over a certain period to warn the driver about possible collisions. In the prediction system, the accuracy of the vehicle's lateral position is crucial, as false positive or false negative warnings may occur depending on the lane width and each vehicle size, even if the position accuracy error is within 1 meter.

The likely position of the vehicle after a certain period can be roughly classified into one of nine categories, conveniently represented as a grid in Figure 5. The prediction system consists of a lane change prediction model to determine the vertical lines of the grid and a trajectory prediction model to determine the horizontal lines.

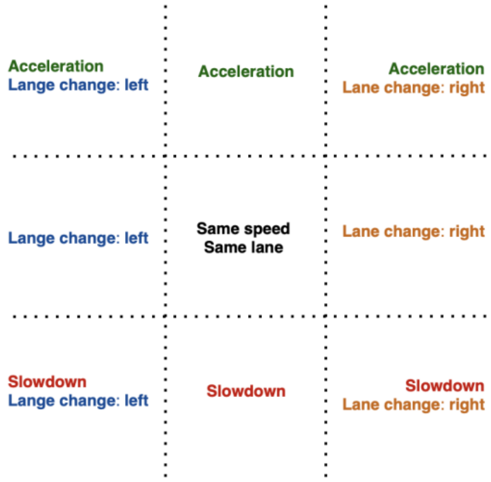

Figure 5. Grid of possible vehicle position at time $t + 1$.

To avoid overfitting in the prediction model, it is essential to select input data that has a high correlation with the output data. The correlation coefficient characterizes the linear dependency between parameters. The coordinates and size of the bounding box obtained from the camera have a high correlation with the relative position between the host vehicle and the neighboring vehicle. Additionally, speed, longitudinal acceleration, heading, and steering angle also correlate with the relative position. Therefore it is expected that the input data values used for the prediction model will be (6)–(8).

$$X = [x_{bbox,bsm}^{t-(h-1)}, \dots, x_{bbox,bsm}^{t-1}, x_{bbox,bsm}^{t}] , \tag{6}$$

$$x_{bbox}^{t} = [xmin, xmax, ymin, ymax, width, height] , \tag{7}$$

$$x_{bsm}^{t} = [x^t, y^t, v^t, \alpha^t, \theta^t, \delta^t] , \tag{8}$$

where $xmin, xmax, ymin, ymax$ – coordinates of the bounding box;
x, y – coordinates of the neighboring vehicle relative to host vehicle;
$v, \alpha, \theta, \delta$ - speed, acceleration, heading, and steering angle, respectively;
h – time of the trajectory observation.

The output of the prediction system includes predicted coordinates of the surrounding vehicle (9)–(10).

$$Y = [y_{position}^{t+1}, y_{position}^{t+2}, \dots, y_{position}^{t+p}] , \tag{9}$$

$$y_{position}^{t} = [x, y] , \tag{10}$$

where x, y – coordinates of the neighboring vehicle relative to host vehicle;
p – prediction time.

6. Research results

Perception System.

The use of the random forest algorithm is investigated for the lane change prediction. It is an ensemble machine learning method that constructs numerous decision trees during model training and produces the mode for the classes (classifications) of the constructed trees through majority voting [15]. Each decision tree has a high variance complexity, but the ensemble reduces variance through bagging and lowers the risk of overfitting. The four steps of the algorithm include:

1. generate random sample with replacement from the dataset: some examples will appear multiple times, while approximately one-third of the examples will not be included;
2. select random input features subset;
3. construct a decision tree that classifies the examples in the sample; the feature for splitting is selected at each node of the tree from a randomly chosen subset of features; the best feature for splitting is selected based on the information gain criterion;
4. repeat steps 1-3 to construct more decision trees.

In general, the more decision trees in the ensemble, the higher the computational expenses, but the better the classifier's performance. If there are factors that significantly affect the results, the algorithm provides high classification performance. The driver's intention to change lanes is closely related to the vehicle's lateral position, making the lateral movement of the bounding box obtained from the camera sensor an important factor in the prediction model. Additionally, steering angle data and accurate heading angle obtained through V2V communication are also important factors in lane change prediction models.

Many existing studies consider lane change prediction using machine learning algorithms [16, 17]. Since the crucial factors can be obtained through the camera and V2V communication, this study employs the random forest algorithm to predict the intention of a lane change.

The lane change process continues for time d depending on driver's behavior and finishes at point T_f at which the vehicle reaches the centerline of the target lane. The start point of the lane change T_s can be determined as the moment that lags behind T_f by d, as demonstrated in Figure 6. The training data labels are then set accordingly based on the state of the vehicle: lane keeping (0), lane change to the left (1), and lane change to the right (2).

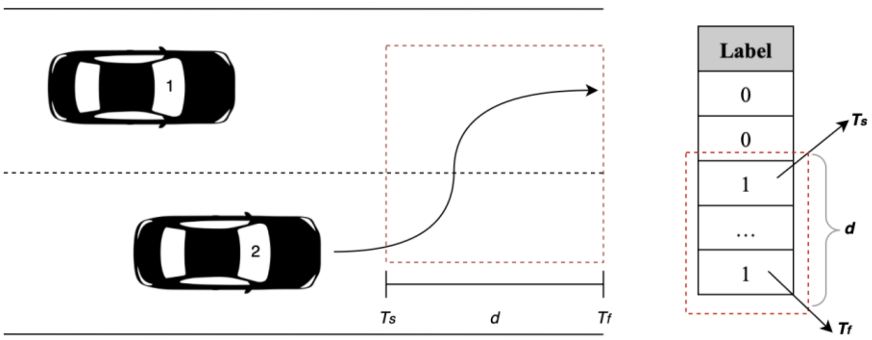

Figure 6. Lane change process in data labeling.

Prediction System.

Machine learning algorithms for supervised learning typically assume that input data is independent and identically distributed. However, for vehicle trajectory prediction, the input data is presented as a time series which is not independent. Considering this, the proposed trajectory prediction model uses a recurrent neural network (RNN) that handles sequentially structured data as input. RNNs have a drawback in the form of exponential growth or decay of the gradient for long sequences of input data. Therefore a Long Short-Term Memory (LSTM) neural network is usually used instead, which is stable and mitigates the typical problems of traditional RNNs [18].

The LSTM network consists of LSTM modules – recurrent network modules capable of remembering values over both short and long periods. LSTM modules are grouped into blocks. An LSTM block contains three gates that control the flow of information at the inputs and outputs of the block's memory, determining which part of its memory is retained and which is overwritten. By using these gates, LSTM networks effectively manage long-term dependencies and maintain stable learning processes, making them well-suited for the complex task of vehicle trajectory prediction.

The gates are implemented as logistic functions to calculate values in the range [0,1]. Multiplying the signal by these values allows partial allowance or prohibition of the flow at the memory's input and output. The three gates are the input gate (i_t), the output gate (o_t), and the forget gate (f_t), depicted in Figure 7 [19].

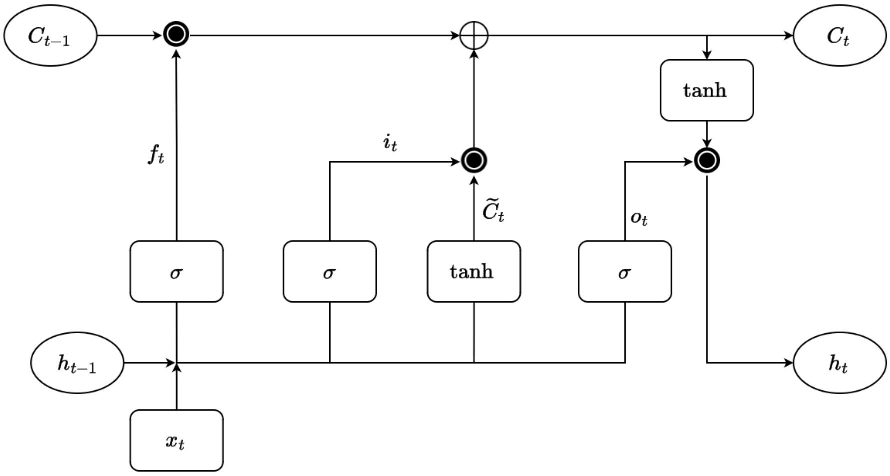

Figure 7. LSTM neural network architecture.

The forget gate f_t determines which information needs to be forgotten using the current input data x_t and the hidden state from the previous time step h_{t-1}.

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)\ , \tag{11}$$

where x_t – input date vector;
h_{t-1} – hidden state from the previous time step;
W_f, U_f – weight matrices of the forget gate for the input and hidden state respectively;
b_f – bias vector of the forget gate;
σ – sigmoid activation function.

The input gate i_t controls the extent to which new information enters the memory, essentially the weight of the new information being received.

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i)\ , \tag{12}$$

$$\hat{C}_t = tanh(W_c x_t + U_c h_{t-1} + b_c)\ , \tag{13}$$

where $\hat{C}_t$ – proposes a new candidate cell state;
W_i, U_i – weight matrices of the input gate for the input and hidden state respectively;
b_i – bias vector of the input gate;
σ – sigmoid activation function;
$tanh$ – hyperbolic tangent activation function.

The cell state is updated according to equation (14).

$$C_t = f_t \times C_{t-1} + i_t \times \hat{C}_t\ , \tag{14}$$

where C_t– new cell state;
f_t – forget gate, determining how much of the previous cell state C_{t-1} should be retained;

i_t – input gate, determining how much of the new candidate cell state $\hat{C}_t$ should be added.

Then the hidden state h_t is calculated based on the cell state C_t and the output gate o_t. The output gate o_t controls the extent to which the value stored in the memory is used to compute the block's output activation function. This process is defined by equations (15)–(16).

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o)\,, \tag{15}$$

$$h_t = o_t \cdot tanh(C_t)\,, \tag{16}$$

where W_o, U_o– weight matrices of the output gate for the input and hidden state respectively;
h_{t-1}– hidden state from the previous time step;
b_o – bias vector of the output gate;
σ – sigmoid activation function;
$tanh$ – hyperbolic tangent activation function.

The memory in an LSTM network is based on a recurrent neural network and is divided into encoding and decoding parts [7]. Initially, the input value is received by the encoding part, and a corresponding vector is created. During decoding, the obtained vector is used for the recursive generation of the output value, as shown in Figure 8.

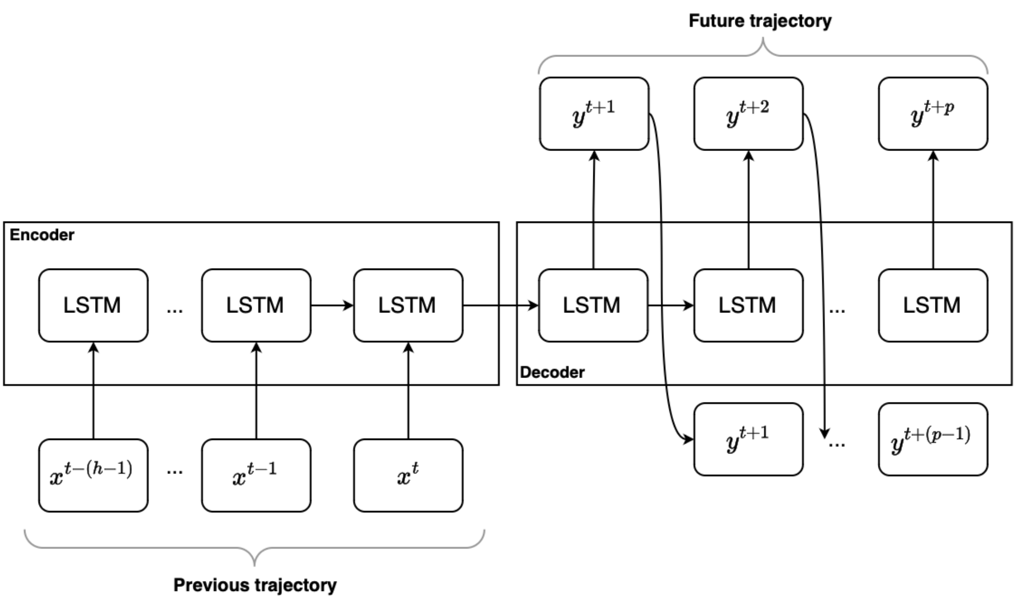

Figure 8. Encoding and decoding components of LSTM.

7. Prospects for further research development

As part of the research, the software code for vehicle trajectory prediction and lane change prediction was developed and implemented. Currently, all developed algorithms have been tested on a small set of simulated data, confirming their functionality. The next crucial step in the research is to collect a dataset to test the developed algorithms in a real environment, which will allow for evaluating their effectiveness and accuracy under complex road conditions.

The further steps include:

1. collecting and preparing real data for training and testing the models;
2. evaluating the performance of the developed algorithms on real data;
3. improving the models based on the results obtained and adapting them to various road conditions;
4. integrating the developed systems into modern autonomous driving platforms to enhance road safety and traffic efficiency.

8. Conclusions

This work analyzes a vehicle trajectory prediction system that utilizes cameras, laser LiDAR scanning, and V2V communication. By utilizing the YOLO algorithm for object detection and differential GPS for precise positioning, the proposed system achieves high accuracy in real-time vehicle positioning. The additional use of the Random Forest algorithm, an ensemble machine learning method that constructs numerous decision trees, allows for predicting the driver's intent to change lanes. Meanwhile, the application of the Long Short-Term Memory (LSTM) neural network ensures accurate trajectory prediction, enhancing the safety and efficiency of autonomous vehicles. This research has laid the foundation for further development in trajectory prediction and lane change prediction, which are essential components for the advancement of autonomous vehicles and driver assistance systems, particularly collision avoidance systems.

References:

1) Bengler, K., Dietmayer, K., Farber, B., Maurer, M., Stiller, C., Winner, H. (2014). Three Decades of Driver Assistance Systems: Review and Future Perspectives. *IEEE Intelligent Transportation Systems Magazine*, *6*(4), 6–22. https://doi.org/10.1109/mits.2014.2336271

2) Lefèvre, S., Vasquez, D., Laugier, C. (2014). A survey on motion prediction and risk assessment for intelligent vehicles. *ROBOMECH Journal*, *1*(1). https://doi.org/10.1186/s40648-014-0001-z

3) Altche, F., de La Fortelle, A. (2017). An LSTM network for highway trajectory prediction. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE. https://doi.org/10.1109/itsc.2017.8317913

4) Deo, N., Trivedi, M. M. (2018). Multi-Modal Trajectory Prediction of Surrounding Vehicles with Maneuver based LSTMs. In *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE. https://doi.org/10.1109/ivs.2018.8500493

5) Zyner, A., Worrall, S., Nebot, E. (2020). Naturalistic Driver Intention and Path Prediction Using Recurrent Neural Networks. *IEEE Transactions on Intelligent Transportation Systems*, *21*(4), 1584–1594. https://doi.org/10.1109/tits.2019.2913166

6) Kim, B., Kang, C. M., Kim, J., Lee, S. H., Chung, C. C., Choi, J. W. (2017). Probabilistic vehicle trajectory prediction over occupancy grid map via recurrent neural network. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE. https://doi.org/10.1109/itsc.2017.8317943

7) Park, S. H., Kim, B., Kang, C. M., Chung, C. C., Choi, J. W. (2018). Sequence-to-Sequence Prediction of Vehicle Trajectory via LSTM Encoder-Decoder Architecture. In *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE. https://doi.org/10.1109/ivs.2018.8500658

8) Thiemann, C., Treiber, M., Kesting, A. (2008). Estimating Acceleration and Lane-Changing Dynamics from Next Generation Simulation Trajectory Data. *Transportation Research Record: Journal of the Transportation Research Board*, *2088*(1), 90–101. https://doi.org/10.3141/2088-10

9) Tan, H.-S., Huang, J. (2006). DGPS-Based Vehicle-to-Vehicle Cooperative Collision Warning: Engineering Feasibility Viewpoints. *IEEE Transactions on Intelligent Transportation Systems*, *7*(4), 415–428. https://doi.org/10.1109/tits.2006.883938

10) Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. https://doi.org/10.1109/cvpr.2016.91

11) Xiang, W., Gozalvez, J., Niu, Z., Altintas, O., Ekici, E. (2009). Wireless Access in Vehicular Environments. *EURASIP Journal on Wireless Communications and Networking*, *2009*(1). https://doi.org/10.1155/2009/576217

12) Kenney, J. B. (2011). Dedicated Short-Range Communications (DSRC) Standards in the United States. *Proceedings of the IEEE*, *99*(7), 1162–1182. https://doi.org/10.1109/jproc.2011.2132790

13) Morsdorf, F., Meier, E., Kötz, B., Itten, K. I., Dobbertin, M., Allgöwer, B. (2004). LIDAR-based geometric reconstruction of boreal type forest stands at single tree level for forest and wildland fire management. *Remote Sensing of Environment*, *92*(3), 353–362. https://doi.org/10.1016/j.rse.2004.05.013

14) Wojke, N., Haselich, M. (2012). Moving vehicle detection and tracking in unstructured environments. In *2012 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. https://doi.org/10.1109/icra.2012.6224636

15) Random Decision Forests. (2017). In *Encyclopedia of Machine Learning and Data Mining*(p. 1054). Springer US. https://doi.org/10.1007/978-1-4899-7687-1_100391

16) Dogan, U., Edelbrunner, J., Iossifidis, I. (2011). Autonomous driving: A comparison of machine learning techniques by means of the prediction of lane change behavior. In *2011 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE. https://doi.org/10.1109/robio.2011.6181557

17) Motamedidehkordi, N., Amini, S., Hoffmann, S., Busch, F., Fitriyanti, M. R. (2017). Modeling tactical lane-change behavior for automated vehicles: A supervised machine learning approach. In *2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*. IEEE. https://doi.org/10.1109/mtits.2017.8005678

18) 5. Deep Learning: RNNs and LSTMs. (2020). In *Angular and Deep Learning Pocket Primer*(pp. 167–192). De Gruyter. https://doi.org/10.1515/9781683924715-006

19) Gers, F. A., Schmidhuber, J., Cummins, F. (2000). Learning to Forget: Continual Prediction with LSTM. *Neural Computation*, *12*(10), 2451–2471. https://doi.org/10.1162/089976600300015015