IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 8 October 2023, accepted 18 October 2023, date of publication 25 October 2023, date of current version 2 November 2023.

*Digital Object Identifier 10.1109/ACCESS.2023.3327433*

RESEARCH ARTICLE

# Surrounding Vehicle Motion Prediction in Non-Lane-Based Environments for ADAS Enhancement

**JOSEPH ANTONY AND SUCHETHA M**
School of Electronics Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu 600 127, India
Centre for Healthcare Advancement, Innovation and Research, Vellore Institute of Technology, Chennai, Tamil Nadu 600 127, India
Corresponding author: Suchetha M (suchetha.m@vit.ac.in)

**ABSTRACT** The significance of forecasting the behavior of road agents is on the rise, particularly in Advanced Driver Assistance Systems (ADAS) enhancement. Predicting road agents' intentions holds paramount importance for Autonomous vehicles, especially considering the forthcoming coexistence of ADAS systems with heterogeneous road entities within urban roadways. The behavioral attributes of non-lane based traffic, featuring a mix of various elements, are prevalent not only in urban scenarios but also in unstructured environments. This research aims to predict the movement patterns of surrounding vehicles in non-lane based environments. This study captures the surrounding vehicles' intent to utilize tight lateral spaces in non-lane based environments through the variations of lateral descriptor values. The investigation takes into account several factors, including leveraging contextual cues, retaining spatial data related to neighboring vehicles, and identifying driving patterns. To achieve these objectives, a hybrid model is introduced, combining a modified structured Long Short Term Memory (LSTM) with lateral descriptor based uncertainty estimation on top of established detection and tracking algorithms. This integration enhances the ability to capture spatial attributes of neighboring vehicles along with the assessment of traffic conflict indicators with lateral descriptor contextual cues. The methodology is evaluated across two distinct datasets: one simulating scenarios of neighboring vehicles within well-defined urban road setups, and the other representing non-structured environments. The empirical findings highlight the effectiveness of the proposed method, showcasing an impressive 24.69% enhancement in prediction accuracy compared to baseline models with 5-seconds prediction horizon.

**INDEX TERMS** ADAS, heterogeneous traffic, LSTM, mixed traffic, lateral descriptor, non lane-based.

## I. INTRODUCTION

Three major factors to achieve ADAS systems on urban roads are road scene awareness, understanding other road agents' intentions and predicting their future motions. Human drivers analyze road scene, understand the intentions of other road users from their trajectory and predict their future trajectories based on experience [1]. ADAS systems find major breakthrough developments with latest sensors, high performance computing platforms and good connectivity. They witnessed number of success in highway traffic environments scenarios, however they are evolving to achieve real-time performance on edge devices with limited computing power [2]. Also, ADAS systems face numerous challenges in urban traffic environments and with non-lane-based environments [3]. The characteristic differences between structured and non-structured environments have been explained in detail in [4]. The differences between lane-based traffic and non-lane based traffic can be observed with respect to the type of road users, driving discipline and movement directions. In lane-based environments, mostly the driving is disciplined, and all surround vehicles will be moving in same direction as that of subject vehicle. However, in non-lane-based environments, the driving is widely undisciplined and concurrent appearance of various classes such as two wheeler, three wheeler, pedestrians

The associate editor coordinating the review of this manuscript and approving it for publication was R. K. Saket.

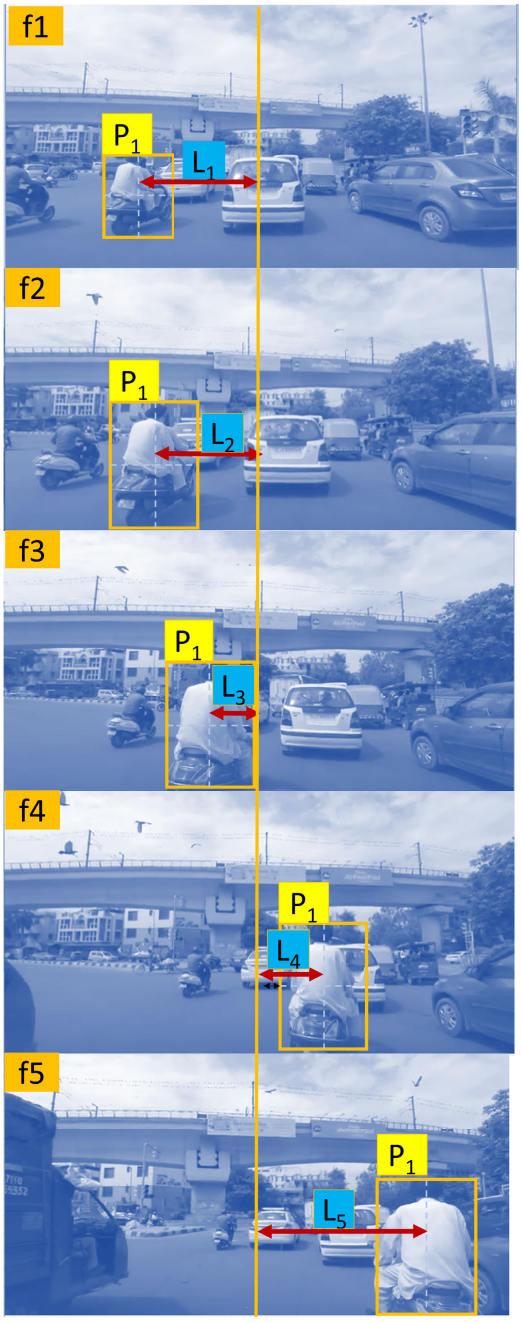


**FIGURE 1.** Typical non-lane-based scenario and lateral descriptor changes.

and stray animals can be present. In lane-based scenarios, vehicles will be confined to the lane and hence the leading vehicle is referred as ego vehicle. Whereas in non-lane-based scenarios, vehicles moving in adjacent lane, suddenly change the course and gets into the track of subject vehicle to utilize the available lateral space and hence referred as surround vehicles. Adaptation of current ADAS systems to non-lane-based environments hugely rely on intent forecast and motion trajectory prediction. A typical scenario in non-lane-based environment is explained in Figure 1, where a two-wheeler rider moves from left edge to right edge in successive frames. Apparently, the changes in lateral position values and bounding box dimensions of this two-wheeler provide critical information on lateral space utilization. In non-lane-based scenarios, lateral descriptors play a vital role in Time to Collision (TTC) calculation. Lateral descriptor is the lateral distance between mid-point of the subject vehicle and lead vehicle, and they are also referred as Center Lane Separation (CS). A CS scenario in a non-lane-based environment is explained in Figure 2. In this scenario, few vehicles moving in front of the subject vehicle. The CS between subject vehicle and surround vehicle 1 ($SV_1$) is $CS_1$ and that of Surround vehicle 2 ($SV_2$) is $CS_2$. The TTCs corresponds to $SV_1$ and $SV_2$ are different and hence different control actions might be required based on lateral descriptor value changes.

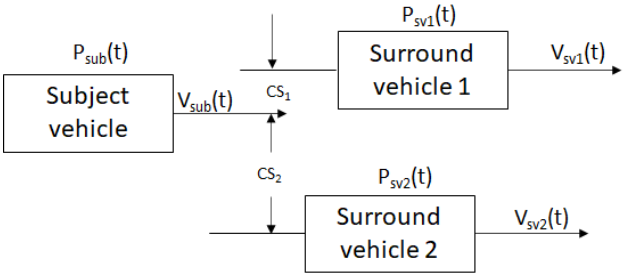


**FIGURE 2.** Lateral distancing between subject vehicle and surround vehicles.

Another scenario with varying lateral position values, varying bounding box dimensions and the bottom edge movement of tracked object is explained in Figure 3. These visual cues provide information about the target vehicle's dynamics and possibility of conflict occurrence. Therefore, a hybrid model was proposed that uses lateral descriptor contextual cues in conjunction with pattern based modified LSTM approach. This model combines the advantages of retaining surrounding vehicles spatial information and their interactions for trajectory prediction.

The main contributions of this paper can be summarized as follows. A modified Multivariate multiple parallel structured LSTM network is proposed for trajectory prediction. This network is specially adapted to manage input sequences centered around surround vehicle centroids, enabling the prediction of their future coordinates. Following the prediction phase, the model estimates uncertainty based on lateral descriptor contextual cues, an inventive approach to calibrate traffic conflict indicators, particularly tailored for non-lane-based scenarios. This lateral descriptor-based method achieves accurate prediction of surrounding vehicles future positions. This novel methodology enables the evaluation of TTC from onboard camera feed with moving reference points in contrast to conventional TTC calculation from fixed reference points.

The rest of this paper is organized as following. Section II presents a brief review of the related work for the problem. Section III describes the proposed solution. Section IV

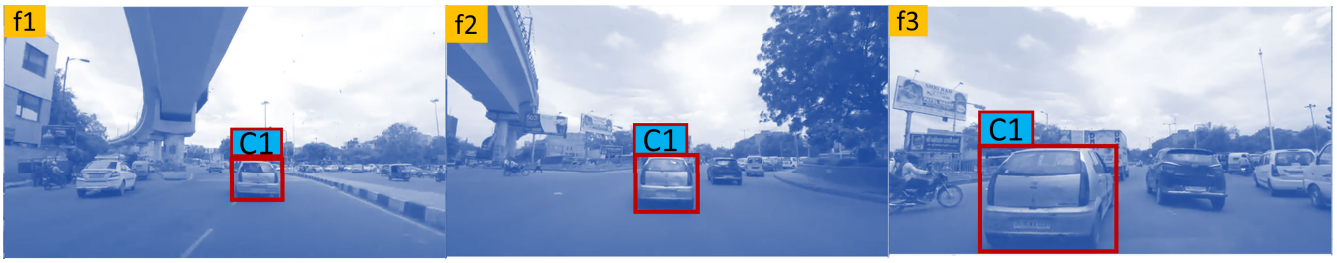


**FIGURE 3.** Vehicle tracking scenarios with appearance vector.

describes the datasets used, and Section V explains validation approach and the results. Section VI concludes this paper.

## II. RELATED WORK

Motion trajectory Prediction approaches are categorized into Physics based, Planning based, Pattern based and Contextual cue based approaches as summarized in Table 1. Physics-based trajectory predictions work well for lane based scenarios and for non-lane-based scenarios pattern-based trajectory prediction produce better results.

Trajectory Prediction approaches began with Kalman filter applications such as Extended Kalman Filter and Unscented Kalman Filter. Kalman filter approach is a typical example of physics-based approach. Physics based approaches suffer with the inability to adapt to driving situations. Planning based approaches consider the reasoning about end goals. Since the goals are incorporated into the prediction, these approaches perform better than physics-based approaches. However, planning based approaches do not model the interaction between neighboring agents. Contextual cue based approaches overcome this constraint by incorporating neighborhood interactions. Recently pattern-based approaches such as LSTM networks are widely used for trajectory prediction applications because of the accuracy and speed. LSTM approach is capable of retaining vital information over a long-time sequence. This approach captures inter dependencies of multiple agents and preserve the spatial information of neighbors. Also, LSTM networks are best suited for mono vision camera applications. From the literature summarized above the trajectory prediction methods based on single approach performs well on the focused criteria but they fail to cover other aspects impacting the trajectory prediction. And also, these methods do not take into account the parameters specific to non-lane-based environments.

Among various ADAS functions, Adaptive cruise control (ACC) is a complex feature that works both in highway environments and in urban environments as well. Partial automation of vehicle's longitudinal control with respect to the ego-vehicle's dynamics is achieved through ACC. ACC uses onboard sensors to follow the preceding vehicle with optimal headway spacing that improves the safety, comfort, and stability of the vehicle [29]. With this function, speed of the subject vehicle is maintained as per driver's input and at the same time a safe distance is maintained with respect to the lead vehicle. To develop functions like ACC, accurate measurement of conflict indicators such as TTC is important. There are various references available to calculate traffic conflict indicators from a stationary camera feed where the camera is stationary and located close to traffic signals or intersections. However, functions like ACC require TTC calculations from onboard sensors and onboard camera feeds.

Traffic conflict study is a proactive approach that helps to avoid crashes before they occur. Traffic conflicts could be measured by multiple evaluation indicators such as TTC [30], Time to Accident (TA) [31], Deceleration Rate to Avoid Collision (DRAC) [32], and Post Encroachment Time (PET) [33]. Though there are various temporal safety indicators used for different applications, PET and TTC have been widely considered as potential parameters in defining warning threshold for ACC/collision avoidance system [34]. Due to its simplicity and reliability, usage of TTC is widely used for different traffic situations specially to analyze and extract high-risk lane change interaction patterns as given in [35]. A minimum TTC value $TTC_{min}$ is an indicator to estimate the criticality of an encounter. In short, lower the $TTC_{min}$ threshold, higher will be the severity of a collision. Most of the research on TTC in traffic safety evaluation has been focused on lane-based traffic scenarios. Different $TTC_{min}$ thresholds have been defined to differentiate critical behavior and normal behavior [36], safe and unsafe vehicle encounters [37]. Selection of thresholds for PET and TTC considering different traffic dynamics, geometric designs, and speed limit are explained in [38]. These thresholds help to provide an early collision avoidance warning to the drivers when TTC falls below the recommended/defined threshold. TTC also helps to characterize the collision risk as low, moderate, and high [39], [40]. TTC calculation for non-lane-based environments require the observation on lateral descriptor contextual cues. Das and Maurya [41] has made an important observation that vehicles in lane-based traffic maintain a mean distance of 1.78m from the left edge of the lane, with 0.34m standard deviation. This observation sets a threshold criterion to distinguish lateral offset for lane based and non-lane-based traffic. A detailed insight into TTC threshold calculations based on type of lead vehicle and the correlation between TTC and center line separation with copula literature framework was given by this

**TABLE 1. Existing approaches comparison.**

| Approach | Technique | Advantages | Limitations |
|---|---|---|---|
| Physics based | Kalman Filter [5]<br>Constant Velocity Model [6]<br>Constant turn rate velocity model [7]<br>Constant turn rate and Acceleration Model [8]<br>Bicycle Model [9]<br>Intelligent Driver Model [10] | Direct calculation of future motion.<br>Easy Modeling. | Do not adapt to driving situations.<br>Vehicle interactions not considered. |
| Planning based | Forward planning Methods [11],[12]<br><br>Inverse Planning Methods [13] | Considers the reasoning about likely goals. | Complex.<br><br>Increased computations. |
| Context cue based | Coupled Hidden Markov Model (HMM) [14]<br>double- layer HMMs [15]<br>Rule based models [16]<br>Lateral descriptors [17] | Considers the interaction between neighboring vehicles | Prediction algorithm is complex |
| Pattern based | Gaussian mixture model [18],<br><br>logistic regression [19]<br>Bayesian networks [20]<br>Markov chain [21]<br>Support Vector Machine [22]<br>HMM [23]<br>Recurrent Neural Network (RNN) [24]<br>LSTM [25],[26],[27],[28] | Considers both driver intention and target vehicles states. | Vehicle interactions not considered. |

research. Typical distribution profiles of best fitted stochastic uncertainties models for lateral descriptors and TTC are presented in Figure 4. Logistic distribution and normal distribution were selected for TTC and lateral descriptors respectively to provide the best fit. The histogram plots indicate the TTCs between surround vehicles range from 0s to 3.6s while the peak of distribution lies at 1.5s. Similarly, the lateral descriptor ranges up to 3m and the peak occurs at 1.5m. From this figure it is evident that the variations in longitudinal gaps and lateral descriptors provide a better understanding of surrounding objects in non-lane-based traffic streams and there exists a strong relationship between them.

From the literature cited above, it is evident that a dominant pattern-based prediction approach is necessary to capture spatial information of surrounding vehicles. However, this approach alone will not be sufficient for non-lane-based environments, because the road agents in these environments often change their course and hence the model needs to take into account the parameters critical for non-lane-based environments.

## III. PROPOSED METHOD

In a scene with concurrent road users utilizing tight lateral road spaces, the problem of estimating vehicle conflict indicators is of twofold. The first challenge is the appropriate network that effectively considers the tight lateral vehicle interactions. The next challenge is to factor the effect of lateral descriptors to accurately predict the TTC on top of prediction. Based on these premises, this paper aims to provide a framework with a modified encoder decoder structural LSTM network to predict the positions of the surrounding vehicles that are being tracked. In addition, this framework models the dependence between longitudinal descriptors and lateral descriptors depending upon the lead vehicle and enables the accurate calculation of respective TTC. Framework for the proposed method is given in Figure 5. The main stages of this framework are detection, tracking, trajectory prediction, CS calculation, TTC estimation and conflict prediction. These stages are explained in the following sections.

### A. SURROUNDING VEHICLES DETECTION

Onboard camera feeds are sliced into frames and the frames are fed into You Only Look Once (YOLO) framework which is the popular network used for object detection [42] due to their speed, accuracy, easy to train, validate and deployment. YOLO Model combines bounding box prediction with class labels in an end-to-end differentiable network. YOLO models consist of 3 main stages. First stage is a convolutional neural network which extracts image features with different granularity. Second stage consists multiple layers to mix and combine image features and pass to prediction stage. Third stage consumes those features and predict the bounding boxes and classes. YOLO_V5 is used in the proposed framework for the object detection as it is a performance improvement version using PyTorch training procedures. The input image is divided into an SxS grid and for each grid YOLO predicts multiple bounding boxes. In addition to bounding box center coordinates(x,y), width(w) and height(h), YOLO also predicts the probability of each object class(C). The absolute position of a predicted bounding box is calculated using the following equations.

$$b_x = \sigma(t_x) + c_x \tag{1}$$
$$b_y = \sigma(t_y) + c_y \tag{2}$$
$$b_w = p_w \cdot e^{t_w} \tag{3}$$
$$b_h = p_h \cdot e^{t_h} \tag{4}$$

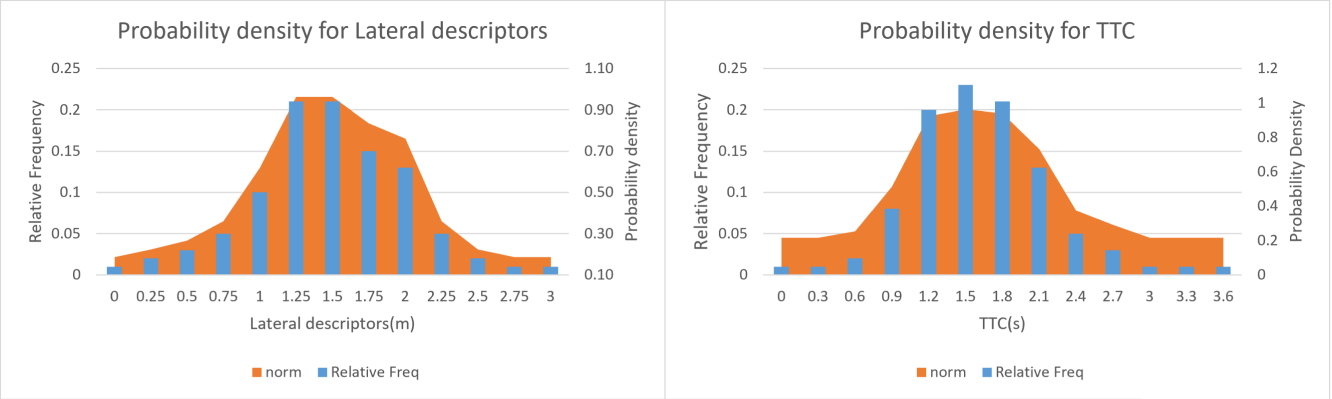


**FIGURE 4.** Probability density for lateral descriptors and TTC.

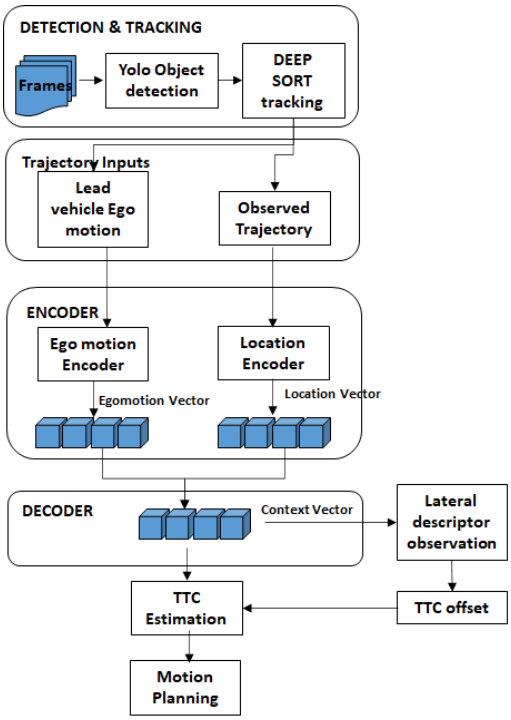


**FIGURE 5.** Proposed framework.

where $(b_x, b_y)$ represents the predicted center coordinates of the bounding box and $(c_x, c_y)$ represents the coordinates of the top-left corner of the grid cell. The dimensions of the anchor boxes are represented by $(p_w, p_h)$ and $(t_x, t_y, t_w, t_h)$ represents the predicted offsets and dimensions. Softmax activation is used to calculate the class probabilities for an object in a grid cell.

$$P(class|obj) = \frac{e^{s_{class}}}{\Sigma_{i=1}^{c} e^{s_i}} \tag{5}$$

where $s_{class}$ represents the raw score for a specific class '$c$'.

### B. SURROUNDING VEHICLES TRACKING

DeepSORT framework is used for surround vehicles tracking like the one used in [43]. DeepSORT is an extension of the SORT (Simple Online Realtime Tracking) algorithm. Kalman filter is the main component in DeepSORT algorithm for noise factoring and uses prior state to predict the good fit for bounding boxes. This algorithm also uses 'Appearance' metric to overcome the shortfalls of Kalman filter like occlusions and different viewpoints. Deep Sort method is suitable for approaches where the data is used on the fly [44]. The computational resources required for this framework is less compared to other popular deep learning approaches. The prediction step state and Error covariance are denoted by

$$\hat{x}_{k|k-1} = F_k \hat{x}_{k-1|k-1} + B_k u_k \tag{6}$$
$$P_{k|k-1} = F_k P_{k|k-1} F_k^T + N_k \tag{7}$$

The update step state and Error covariance are denoted by

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + G_k(Z_k - H_k \hat{x}_{k|k-1}) \tag{8}$$
$$P_{k|k} = (I - G_k H_k) P_{k|k-1} \tag{9}$$

where the Kalman Gain $G_k$ is calculated by

$$G_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + M_k)^{-1} \tag{10}$$

Here $F_k$ represents state transition matrix, $B_k$ represents control input matrix and $N_k$ represents process noise. Similarly, $Z_k$ represents measurement matrix, $H_k$ represents observation matrix and $M_k$ represents measurement noise.

### C. SURROUNDING VEHICLE MOTION PREDICTION

Multivariate Multi-Step LSTM network is used to generate future trajectory of surround vehicles. There are two types of Multivariate Multi-Step LSTM models to analyze multivariate time series data. They are Multiple Input Multi-Step Output model and Multiple Parallel Input and Multi-Step Output model depending upon the dimension of input sequence and the dimension of predicted output sequence. The problem considered for this work is related to multi-step

time series forecasting because it requires a prediction of multiple time steps into the future. Usually, for multi-step forecasting problems two main types of LSTM models can be used. They are Vector Output Model and Encoder-Decoder Model. Former model outputs a vector directly that can be interpreted as a multi-step forecast and later model is developed for forecasting variable length output sequences. This encoder-decoder model is also known as seq2seq model. The enhanced version of this Encoder-Decoder model for multivariate multi-step with Parallel Input sequences and multi-Step prediction is used in this application.

A typical tracking scenario with 3 frames of varying lateral distance and varying bounding box dimension was shown in Figure 3. Let the observed trajectory is represented as $X = X_1, X_2, X_3, \ldots X_n$. Where $X_i^t = (x_i^t, y_i^t)$ and $x_i^t$, $y_i^t$ corresponds to the observed x, y position of surround vehicle $i$ at time instance $t$. And the predicted trajectory is represented by $Y = Y_1, Y_2, Y_3, \ldots, Y_n$ Where $Y_i^t = (\hat{x}_i^t, \hat{y}_i^t)$ and $(\hat{x}_i^t, \hat{y}_i^t)$ corresponds to the predicted x, y position of surround vehicle $i$ at time instance $t$. Here the observed time series is represented as $t = T_{k-h}, \ldots, T_{k-2}, T_{k-1}, T_k$ and the prediction time series is represented as $t = T_{k+1}, T_{k+2}, \ldots, T_{k+p}$.

$$H_t^{i,(1)} = LSTM^{enc}(H_{t-1}^{i,(1)}, X_t^i; W_H^{(1)}) \tag{11}$$
$$H_t^{i,(2)} = LSTM^{enc}(H_{t-1}^{i,(2)}, H_t^{i,(1)}; W_H^{(2)}) \tag{12}$$
$$H_t^{i,(3)} = LSTM^{enc}(H_{t-2}^{i,(3)}, H_t^{i,(2)}; W_H^{(3)}) \tag{13}$$

where

$$X_t^i = f(x_t^i, y_t^i; W_E) \tag{14}$$

Here f() is the function mapping $(x_t^i, y_t^i)$ to the vector $X_t^i$, $W_E$, $W_H^{(1)}$, $W_H^{(2)}$ and $W_H^{(3)}$ represent weight matrices. $H_t^{i,(k)}$ represent hidden state vector for layers k=1,2,3.

$\hat{Y}_t^i$ is the encoded vector obtained from Eq. 4 and fed to the decoder network.

$$H_t^{i,(1)} = LSTM^{dec}(H_{t-1}^{i,(1)}, \hat{Y}_t^i; W_H^{(1)}) \tag{15}$$
$$H_t^{i,(2)} = LSTM^{dec}(H_{t-2}^{i,(2)}, H_t^{i,(1)}; W_H^{(2)}) \tag{16}$$
$$H_t^{i,(3)} = LSTM^{dec}(H_{t-3}^{i,(1)}, H_t^{i,(2)}; W_H^{(3)}) \tag{17}$$

Prediction output of LSTM decoder for VRU i at time t is given by $Y_t^i = f(\hat{x}_{t-1}^i, \hat{y}_{t-1}^i; W_D)$ and obtained from Eq.8

$$Y_t^i = f(\hat{x}_{t-1}^i, \hat{y}_{t-1}^i; W_D) \tag{18}$$

Here f() is the function mapping $(\hat{x}_t^i, \hat{y}_t^i)$ to the vector $Y_t^i$, $W_D$, $W_H^{(1)}$, $W_H^{(2)}$ and $W_H^{(3)}$ represent weight matrices. $H_t^{i,(k)}$ represent hidden state vector for decoder layers k=1,2,3.

The encoder LSTM and decoder LSTM model are jointly trained to minimize prediction errors.

$$Loss = \sum_{i=1}^{k+p} (Y_i^t - Y(G)_i^t) \tag{19}$$

where $Y_i^t$ and $Y(G)_i^t$ are predicted sequences and ground truth sequences respectively. The architecture of one layer of modified structural LSTM architecture used in this model is given in Figure 6. The layers are dynamically chosen based on the number of vehicles tracked.

### D. TTC CALCULATION

PET is defined as the time difference between the first road user leaving conflict point and second road user reaching the conflict point. When the lead vehicle leaves conflict point at $t_1$ and subject vehicle reaches this point at $t_2$, then PET is defined as follows.

$$PET = t_2 - t_1 \tag{20}$$

TTC is defined as the time remaining for the occurrence of a collision between 2 road users if both road users maintain the course and speed. Let $d_{LV}$ and $speed_{LV}$ represents the distance and speed of lead vehicle. And $d_{SV}$ and $speed_{SV}$ represents the distance and speed of lead vehicle, then TTC is defined as follows.

$$TTC = \left| \frac{d_{LV}}{speed_{LV}} - \frac{d_{SV}}{speed_{SV}} \right| \tag{21}$$

The PET and minimum TTC are calculated using the following three steps. The first step is to start individual counter sequences of the trajectories of subject vehicle and surround vehicles to obtain the conflict point for each lead vehicle. Conflict point shall be obtained if the tracked object area reduction falls below the bounding box dimension change threshold value. The second step is to determine the frame numbers of the pair when approaching this conflict point. The third step is to calculate PET by dividing time difference by frames per second and then to calculate the TTC dynamically.

### E. LATERAL DESCRIPTOR BASED UNCERTAINTY ESTIMATOR MODEL

After TTC is obtained, collision risk is calculated by adding lateral descriptor bias for uncertainty estimation. This lateral descriptor biasing is managed outside LSTM model without affecting iterative training of the neural network. This uncertainty estimation is calculated as follows.

$$LV_{Ui} = \begin{cases} 0, & py(k+1|t) > py(k|t) \wedge A(p(k+1|t)) \leq A(p(k|t)) \\ 1 & py(k+1|t) < py(k|t) \wedge A(p(k+1|t)) >= A(p(k|t)) \end{cases}$$

where $LV_{Ui}$, $k$ and $t$ represent Lead vehicle uncertainty index, prediction step index and time index respectively. $py(k|t)$ and $A(p(k|t)$ represents lateral position and Area of Bounding Box respectively. Similarly, lead vehicle turn direction is calculated as follows.

$$LV_{Td} = \begin{cases} -1, & py(k+1|t) < Y_{Lthres} \\ 0, & py(k+1|t) Y_{Lthres} < py(k+1|t) < Y_{Rthres} \\ 1 & py(k+1|t) > Y_{Rthres} \end{cases}$$

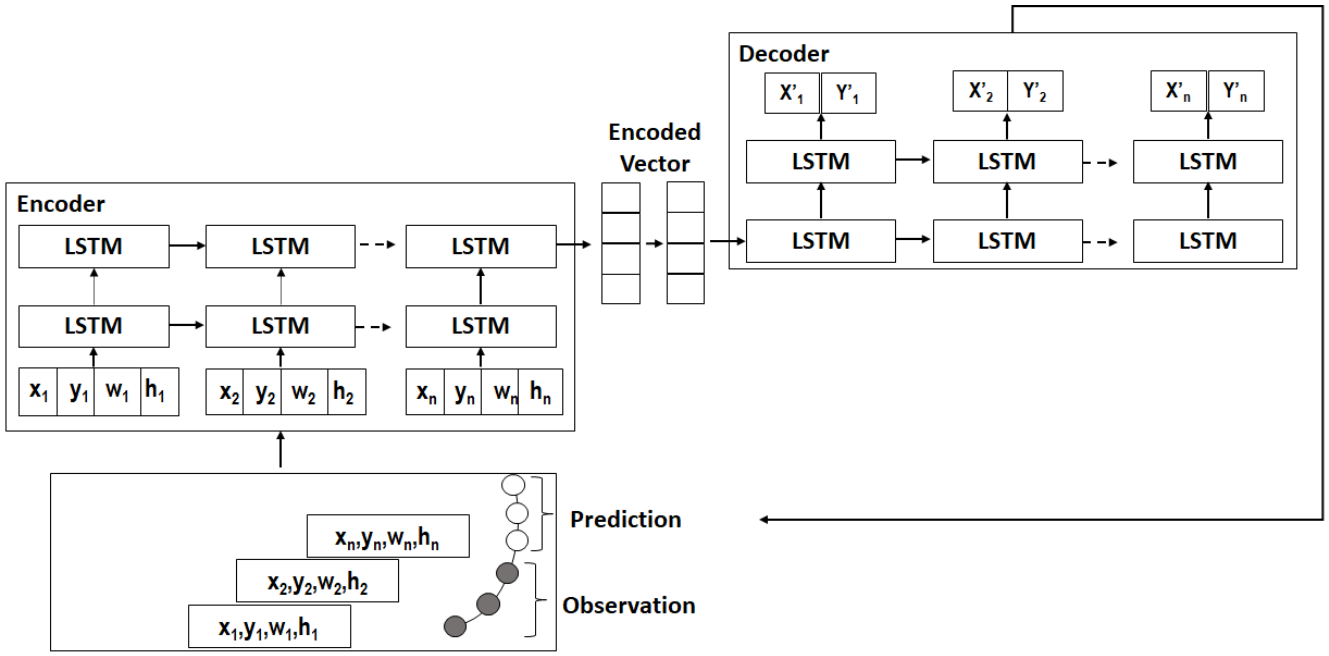


**FIGURE 6.** Structured LSTM network layer for single vehicle.

where $LV_{Td}$, $Y_{Lthres}$ and $Y_{Rthres}$ represent Lead vehicle turn direction, and threshold values for left and right turn scenarios.

There can be 4 possible scenarios from the results obtained above. If $LV_{Ui}$ is 0 or if $LV_{Td}$ is -1 or 0 then those scenarios are considered as non-serious conflicts. On the other hand, if $LV_{Ui}$ is 1 or if $LV_{Td}$ is 1 then those scenarios are considered as serious conflicts for right hand driving environments. Similarly, the uncertainty index can be calculated for Left hand driving environments by changing the threshold criteria.

## IV. DATASET

Two datasets are applied in this work for the accuracy verification. The first dataset is the Next Generation Simulation (NGSIM) dataset [45]. This data set is a widely used dataset to test the prediction accuracy and one such application is explained in [46]. This data set is used to test the Prediction accuracy of the proposed model against 3 baseline models. The baseline models considered here are Dual Learning LSTM Model, Multi modal maneuver-based LSTM Model and Modified LSTM vehicle interactions Model. These models were tested with same NGSIM dataset, and the results are publicly available. This NGSIM dataset is publicly available and derived from the US Federal Highway Administration. Each data slice includes vehicle's parameters, their position, velocity, yaw rate, and size. For this work, the trajectory data from Peachtree Street videos and Lankershim Boulevard video are considered. The second data set was collected from 4 different metro cities in India with mixed heterogeneous traffic. A 1080 pixel on board camera (Model Number: ELP-USBFHD01M) is placed in the windshield to capture the road scenes and the acquired video samples were fed through NVIDIA Jetson TX2 development kit. Video to frames conversion, YOLO object detection algorithms, DEEP SORT tracking algorithms, Encoder Decoder algorithms and TTC estimation algorithms were executed in this NVIDIA board. This data set contains 20446 road elements from about 31923 frames. The actual trajectory of each road element was recorded in the work presented in [4] and this trajectory information was compared against the results obtained with the proposed approach.

## V. IMPLEMENTATION DETAILS

Keras was used to generate in an end-to-end fashion. The number of hidden state dimensions of LSTM encoder and decoder are both 200 and trained with 'ReLU' activation function. The proposed method was trained by Adam optimizer with 0.001 learning rate for 300 epochs with a mini batch size of m=32. The mean squared error (MSE) between the predicted sequence and the ground truth sequence was monitored on the validation loss. The implementation model algorithm is presented in Algorithm 1.

## VI. RESULTS AND DISCUSSION

The efficiency of proposed model against baseline models are listed in Table 2. These baseline models are best performing models to date and their prediction accuracy is high compared to other popular models. These models are selected to verify the accuracy of proposed model with the same NGSIM dataset. The proposed model outperforms the Dual learning baseline model by 14%. The Root Mean Square

**TABLE 2.** RMSE comparison with existing methods.

| Reference | Baseline Model | Prediction Horizon (s) | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| Khakzar et.al[47] | Dual Learning | 0.41 | 0.95 | 1.72 | 2.64 | 3.87 |
| Deo et.al[48] | Multi Modal | 0.58 | 1.26 | 2.12 | 3.24 | 4.66 |
| Dai et.al[49] | Modified LSTM | 0.58 | 1.21 | 1.97 | 2.85 | 3.89 |
| Messaoud et.al[50] | Social Pooling | 0.56 | 1.22 | 2.02 | 3.03 | 4.30 |
| Dai et.al[51] | Convolutional social pooling | 0.61 | 1.27 | 2.09 | 3.10 | 4.37 |
| | Proposed Method | 0.5 | **0.9** | **1.7** | 2.9 | **3.32** |

**TABLE 3.** TTC variations with respect to lateral descriptors.

| CS(m) | 2W | | | 3W | | | Car | | | Truck | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $TTC_{min}$ | $TTC_{max}$ | $TTC_{avg}$ | $TTC_{min}$ | $TTC_{max}$ | $TTC_{avg}$ | $TTC_{min}$ | $TTC_{max}$ | $TTC_{avg}$ | $TTC_{min}$ | $TTC_{max}$ | $TTC_{avg}$ |
| -2 <= CS < -1.5 | 2.4 | 2.9 | 2.65 | 1.2 | 2.5 | 1.85 | 0.024 | 1.03 | 0.527 | 0.024 | 0.048 | 0.036 |
| -1.5 <= CS < -1 | 2.52 | 2.94 | 2.73 | 0.34 | 2.553 | 1.446 | NA | NA | NA | 0.024 | 0.048 | 0.036 |
| -1 <= CS < -0.5 | 2.82 | 2.99 | 2.905 | 0.6 | 1.6 | 1.1 | 0.024 | 2.68 | 1.352 | NA | NA | NA |
| -0.5 <= CS < 0 | 2.64 | 2.88 | 2.76 | 1.78 | 2.01 | 1.89 | NA | NA | NA | NA | NA | NA |
| 0 <= CS < 0.5 | 2.87 | 2.98 | 2.925 | 0.65 | 2.28 | 1.46 | NA | NA | NA | 0.72 | 1.05 | 0.885 |
| 0.5 <= CS < 1.0 | 2.57 | 2.83 | 2.7 | 0.91 | 2.27 | 1.59 | 0.31 | 1.84 | 1.075 | NA | NA | NA |
| 1.0 <= CS < 1.5 | 2.93 | 3.131 | 3.03 | 1.63 | 2.21 | 1.92 | 2.04 | 2.25 | 2.145 | 0.24 | 0.216 | 0.228 |
| 1.5 <= CS < 2 | 2.35 | 2.95 | 2.65 | 1.14 | 2.52 | 1.83 | 0.69 | 1.35 | 1.02 | 0.94 | 2.62 | 1.78 |

**Algorithm 1** Surrounding Vehicle Tracking and Uncertainty Estimation

```
initialize
capture video
for each (frame) do
    num_SVs ← YOLO_V5(frame)
    for n = 1tonum_SVs do
        num_traj ← DEEP_SORT(SV_n)
        for P_h = 1tonum_pred_hori do
            for i = 1tonum_traj do
                X_i, pred ← sub − LSTM_j(X_obs)
                Y_i, pred ← sub − LSTM_j(Y_obs)
                X_Ph+1 = f(X_Ph, N(0, W))
            end for
            i ← i + 1
            calculate (CS)
            calculate (TTC)
            TTC_offset = f(CS, TTC, Veh_Type)
        end for
        P_h ← P_h + 1
    end for
end for
```

Error (RMSE) error of Multimodal LSTM and Modified LSTM model is greater than the proposed method by 28% and 14% respectively. Once the prediction accuracy of this model is verified using NGSIM data set, the second data set which includes the traffic flow from Indian metro cities was applied. This data set is used in the work [4] and it includes all dynamics of non-structured environments. Also, this data set contains trajectories of 2 wheelers and 3 wheelers which are major constituents of multi road agent scenarios.

The TTC variation with respect lateral descriptor values for different lead vehicles are presented in Table 3. The TTC

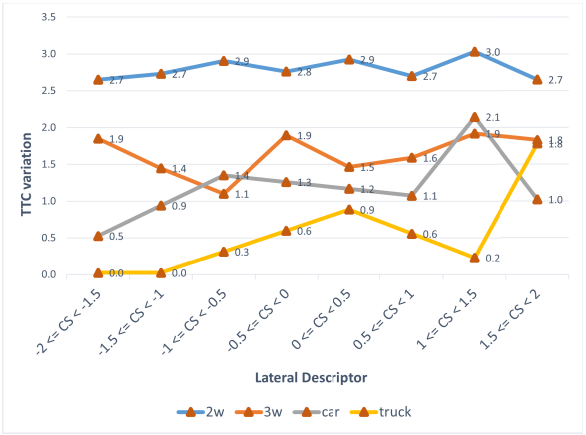


**FIGURE 7.** TTC variation with changing CS vehicle for subject lead pair.

values for different range of CS values are segregated in eight equal intervals. For some range of CS values, this data set do not have the road agent trajectory and those cells are marked as Not available (NA). From this table it is evident that the TTC values around central axis of the frame is slightly less than those around 0.5 m to 1m range. This is because, the lead vehicles in this range of (0.5m <= CS < 1.0m) on both left and right axis tend to divert from the current lane to adjacent lane. However, vehicles perpendicular to central axis have less TTC because those vehicles are either obstructed or maintaining the course with same speed. Again, the TTC for vehicles in the range of (1m <= CS < 2m) on both left and right axis, tend to decrease because the lead vehicles started moving away the subject vehicle lane.

The variation of mean TTC with respect to CS range is given in Figure 7. From this figure it is evident that threshold values for 3W, car and truck are relatively close and the range -1.5m < CS < 1m is crucial to avoid maximum collision instances. Sensitivity graph of the TTC variation for different

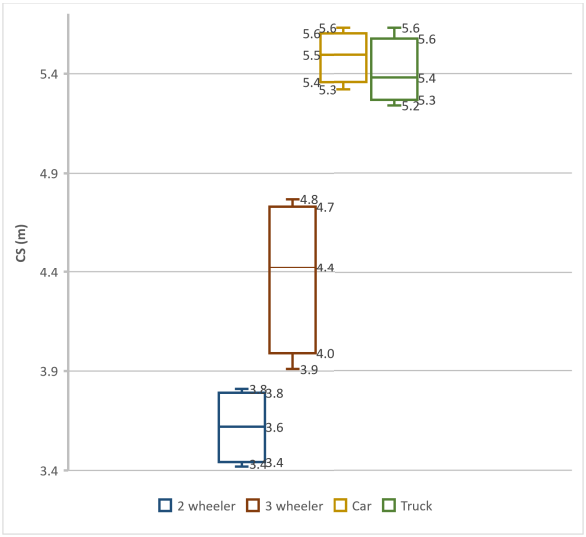


**FIGURE 8.** TTC sensitivity for different road users.

**TABLE 4.** TTC percentage decrease with respect to minimum TTC values.

| TTC at CS=0.34m | 2.7 | 1.71 | 1.6 | 1.57 |
|---|---|---|---|---|
| Lead Vehicle / CS(m) | 2W | 3W | Car | Truck |
| 0.5 | 64.8 | 36.5 | 7.6 | 7.3 |
| 1 | 63.9 | 31.3 | 5.0 | 5.3 |
| 1.5 | 61.6 | 25.0 | 4.2 | 4.3 |
| 2 | 60.8 | 22.5 | 2.3 | 0.4 |

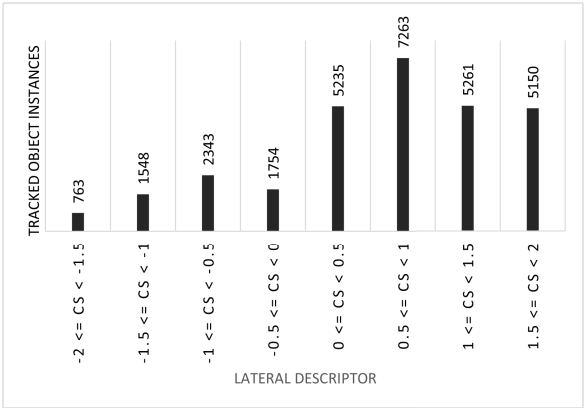


**FIGURE 9.** Trajectories orientation for RWD lanes.

classes of road users are plotted in Figure 8. From this figure, it is evident that the TTC range of 3 wheelers are little high since they are slow moving vehicles compared to other classes. The TTC range for cars and truck are similar because these classes occupy the maximum width of the lane space. The instances of orientation of trajectories with respect to CS range are presented in Figure 9. It is evident that there more instances of trajectories are observed on the right side compared to left side of the central axis. This result is due to the fact that this traffic was observed in the Right-Hand Drive (RHD) scenario. In these driving scenarios

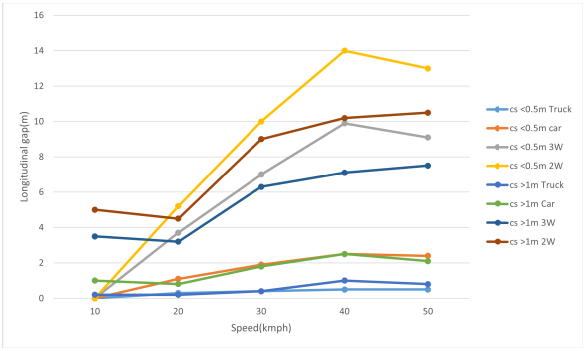


**FIGURE 10.** Longitudinal gap variation with speed.

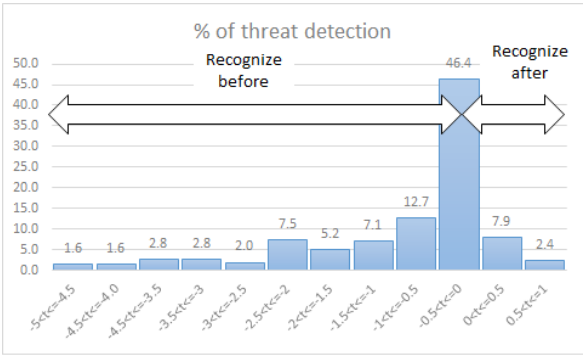


**FIGURE 11.** Percentage of threat detection.

the lead vehicles tend to align to the right-hand side more than the left-hand side direction and vice versa for Left Hand Drive (LHD) scenario. These observations form the basis to conclude which range of CS values are prone for collision events and which direction to central axis will have more lead vehicles and their relationship with driving side such as LHD or RHD. The percentage decrease of TTCs compared with baseline TTC when CS = 0.34m is presented in Table 4. This result matches with the fact that 2 wheelers move fast to utilize tight lateral spaces with more confidence and hence their TTCs are on higher side. Also, the elements in this traffic tend to follow the lead vehicles closely maintaining lower longitudinal gaps. The variations in average longitudinal gap with speed for different lateral descriptors are presented Figure 10. As discussed before, the detected trend further confirms the increasing relationship of longitudinal gap and speed for each lateral descriptor range. With the increase in lateral descriptor between the vehicles, the longitudinal gap decreases at different trail vehicle's speeds. The observation was also made to compare the differences of recognition timing between constant velocity approach and the proposed approach and presented in Figure 11. It is evident that the proposed algorithm recognized the target earlier in around 58% of total cases by up to 5s. Around 36% of cases were identified in same time. Therefore, the proposed algorithm recognized more than 90% of vehicles ahead of time compared to baseline algorithm.

## VII. CONCLUSION

The method proposed in this paper uses tracked object position, dimension, CS variations, bottom edge trajectory to identify the TTC values. This method utilizes LSTM's capability to capture multi agents inter dependencies along with their spatial information and the contextual cues. The relative importance of lateral descriptors and their offset values enables the model to accurately predict surrounding vehicles trajectories depending upon the type of lead vehicle. Both LSTM based prediction and contextual cue-based prediction augments the prediction accuracy with identification of driving patterns along with preserving the spatial information between neighboring vehicles. The proposed model was tested on two video datasets. Experimental results show that the specific range of CS values (0.5m <= CS < 1.0m) are prone for more collision events compared to other ranges. The proposed method can be easily integrated with mono-vision camera feeds which is instrumental to develop ADAS applications for non-structured environments. This method also provides uncertainty estimation from moving references. It is also observed that the direction to central axis will have more lead vehicles depend on the kind of driving environment and this observation will help to confine the focus area to observe critical conflicts. In addition to these parameters, it was also observed that combining lead vehicle brake light illumination with the TTC, would further improve the accuracy of detected TTC. Further investigation needs to be conducted to expand this concept. The proposed model enables calculation of traffic conflict indicators with moving reference points and has the potential to be implemented for mono vision camera applications.

## REFERENCES

[1] Y. Jeong, S. Kim, and K. Yi, "Surround vehicle motion prediction using LSTM-RNN for motion planning of autonomous vehicles at multi-lane turn intersections," *IEEE Open J. Intell. Transp. Syst.*, vol. 1, pp. 2–14, 2020.

[2] D. N. Tran, L. H. Pham, H.-H. Nguyen, T. H. Tran, H.-J. Jeon, and J. W. Jeon, "Universal detection-based driving assistance using a mono camera with Jetson devices," *IEEE Access*, vol. 10, pp. 59400–59412, 2022, doi: 10.1109/ACCESS.2022.3179999.

[3] J. J. Antony and M. Suchetha, "Vision based vehicle detection: A literature review," *Int. J. Appl. Eng. Res.*, vol. 11, no. 5, pp. 3128–3133, 2016.

[4] J. Antony and S. Manikandan, "Expanding vision-based ADAS for non-structured environments," *IET Intell. Transp. Syst.*, vol. 14, no. 6, pp. 620–627, 2020.

[5] T. Qie, W. Wang, C. Yang, Y. Li, Y. Zhang, W. Liu, and C. Xiang, "An improved model predictive control-based trajectory planning method for automated driving vehicles under uncertainty environments," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 4, pp. 3999–4015, Apr. 2023, doi: 10.1109/TITS.2022.3230680.

[6] C. Schöller, V. Aravantinos, F. Lay, and A. Knoll, "What the constant velocity model can teach us about pedestrian motion prediction," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 1696–1703, Apr. 2020, doi: 10.1109/LRA.2020.2969925.

[7] G. Guo and S. Zhao, "3D multi-object tracking with adaptive cubature Kalman filter for autonomous driving," *IEEE Trans. Intell. Vehicles*, vol. 8, no. 1, pp. 512–519, Jan. 2023, doi: 10.1109/TIV.2022.3158419.

[8] L. Zhang, K. Yuan, H. Chu, Y. Huang, H. Ding, J. Yuan, and H. Chen, "Pedestrian collision risk assessment based on state estimation and motion prediction," *IEEE Trans. Veh. Technol.*, vol. 71, no. 1, pp. 98–111, Jan. 2022, doi: 10.1109/TVT.2021.3127008.

[9] Y. Jeong and S. Yim, "Path tracking control with four-wheel independent steering, driving and braking systems for autonomous electric vehicles," *IEEE Access*, vol. 10, pp. 74733–74746, 2022, doi: 10.1109/ACCESS.2022.3190955.

[10] J. Diao, R. Tang, Y. Gu, S. Tian, and Z. Jiang, "Cognitive-digital-twin-based driving assistance," *IEEE Robot. Autom. Lett.*, vol. 8, no. 8, pp. 5188–5195, Aug. 2023, doi: 10.1109/LRA.2023.3291895.

[11] J. Yan, S. Du, and Y. Wang, "Multi-pedestrian tracking in crowded scenes by modeling movement behavior and optimizing Kalman filter," *IEEE Access*, vol. 10, pp. 118512–118521, 2022, doi: 10.1109/ACCESS.2022.3220635.

[12] Q. Wang, B. Ayalew, and T. Weiskircher, "Predictive maneuver planning for an autonomous vehicle in public highway traffic," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 4, pp. 1303–1315, Apr. 2019, doi: 10.1109/TITS.2018.2848472.

[13] T. Chen, C. Guo, H. Li, T. Gao, L. Chen, H. Tu, and J. Yang, "An improved multimodal trajectory prediction method based on deep inverse reinforcement learning," *Electronics*, vol. 11, pp. 40–97, Dec. 2022, doi: 10.3390/electronics11244097.

[14] P. Karle, M. Geisslinger, J. Betz, and M. Lienkamp, "Scenario understanding and motion prediction for autonomous vehicles—Review and comparison," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 16962–16982, Oct. 2022, doi: 10.1109/TITS.2022.3156011.

[15] S. Sun, S. Li, Y. Li, B. Moran, and W. S. T. Rowe, "Smartphone user tracking by incorporating user orientation using a double-layer HMM," *IEEE Trans. Veh. Technol.*, vol. 71, no. 7, pp. 7780–7790, Jul. 2022, doi: 10.1109/TVT.2022.3168142.

[16] R. Bhattacharyya, S. Jung, L. A. Kruse, R. Senanayake, and M. J. Kochenderfer, "A hybrid rule-based and data-driven approach to driver modeling through particle filtering," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 13055–13068, Aug. 2022, doi: 10.1109/TITS.2021.3119415.

[17] B. Gunay, "Methods to quantify the discipline of lane-based-driving," *Traffic Eng. Control*, vol. 44, no. 1, pp. 22–27, 2003.

[18] S. A. Goli, B. H. Far, and A. O. Fapojuwo, "Vehicle trajectory prediction with Gaussian process regression in connected vehicle environment," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 550–555.

[19] Y. Yang and N. Li, "Research on residents' travel behavior based on multiple logistic regression model," *IEEE Access*, vol. 11, pp. 74759–74767, 2023, doi: 10.1109/ACCESS.2023.3297497.

[20] A. Nayak, A. Eskandarian, and Z. Doerzaph, "Uncertainty estimation of pedestrian future trajectory using Bayesian approximation," *IEEE Open J. Intell. Transp. Syst.*, vol. 3, pp. 617–630, 2022, doi: 10.1109/OJITS.2022.3205504.

[21] A. Pirayre, P. Michel, S. S. Rodriguez, and A. Chasse, "Driving behavior identification and real-world fuel consumption estimation with crowdsensing data," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 18378–18391, Oct. 2022, doi: 10.1109/TITS.2022.3169534.

[22] X. Feng, X. Ling, H. Zheng, Z. Chen, and Y. Xu, "Adaptive multi-kernel SVM with spatial–temporal correlation for short-term traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 6, pp. 2001–2013, Jun. 2019, doi: 10.1109/TITS.2018.2854913.

[23] X. Xiong, L. Chen, and J. Liang, "A new framework of vehicle collision prediction by combining SVM and HMM," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 699–710, Mar. 2018, doi: 10.1109/TITS.2017.2699191.

[24] C.-L. Hwang and H. B. Abebe, "Generalized and heterogeneous nonlinear dynamic multiagent systems using online RNN-based finite-time formation tracking control and application to transportation systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 13708–13720, Aug. 2022, doi: 10.1109/TITS.2021.3126662.

[25] Y. Ji, L. Wang, W. Wu, H. Shao, and Y. Feng, "A method for LSTM-based trajectory modeling and abnormal trajectory detection," *IEEE Access*, vol. 8, pp. 104063–104073, 2020, doi: 10.1109/ACCESS.2020.2997967.

[26] Y. Jeong and K. Yi, "Bidirectional long shot-term memory-based interactive motion prediction of cut-in vehicles in urban environments," *IEEE Access*, vol. 8, pp. 106183–106197, 2020, doi: 10.1109/ACCESS.2020.2994929.

[27] L. Hou, L. Xin, S. E. Li, B. Cheng, and W. Wang, "Interactive trajectory prediction of surrounding road users for autonomous driving using structural-LSTM network," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 11, pp. 4615–4625, Nov. 2020.

[28] Y. Feng, T. Zhang, A. P. Sah, L. Han, and Z. Zhang, "Using appearance to predict pedestrian trajectories through disparity-guided attention and convolutional LSTM," *IEEE Trans. Veh. Technol.*, vol. 70, no. 8, pp. 7480–7494, Aug. 2021.
[29] P. Khound, P. Will, and F. Gronwald, "Design methodology to derive over-damped string stable adaptive cruise control systems," *IEEE Trans. Intell. Vehicles*, vol. 7, no. 1, pp. 32–44, Mar. 2022, doi: 10.1109/TIV.2021.3066056.
[30] W. Hu, X. Li, J. Hu, D. Kong, Y. Hu, Q. Xu, Y. Liu, X. Song, and X. Dong, "A safe driving decision-making methodology based on cascade imitation learning network for automated commercial vehicles," *IEEE Sensors J.*, vol. 23, no. 11, pp. 11285–11295, Jun. 2023, doi: 10.1109/JSEN.2023.3256704.
[31] C. Hydn and L. Linderholm, "The Swedish traffic-conflicts technique," in *International Calibration Study of Traffic Conflict Techniques*. Heidelberg, Germany: Springer, 1984, pp. 133–139, doi: 10.1007/978-3-642-82109-7_12.
[32] B. Lv, R. Sun, H. Zhang, H. Xu, and R. Yue, "Automatic vehicle-pedestrian conflict identification with trajectories of road users extracted from roadside LiDAR sensors using a rule-based method," *IEEE Access*, vol. 7, pp. 161594–161606, 2019, doi: 10.1109/ACCESS.2019.2951763.
[33] S. Zhang and M. Abdel-Aty, "Real-time pedestrian conflict prediction model at the signal cycle level using machine learning models," *IEEE Open J. Intell. Transp. Syst.*, vol. 3, pp. 176–186, 2022, doi: 10.1109/OJITS.2022.3155126.
[34] Y. Li, Z. Li, H. Wang, W. Wang, and L. Xing, "Evaluating the safety impact of adaptive cruise control in traffic oscillations on freeways," *Accident Anal. Prevention*, vol. 104, pp. 137–145, Jul. 2017.
[35] Y. Zhang, Y. Zou, Selpi, Y. Zhang, and L. Wu, "Spatiotemporal interaction pattern recognition and risk evolution analysis during lane changes," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 6, pp. 1–11, Jun. 2023, doi: 10.1109/TITS.2022.3233809.
[36] H. Behbahani, N. Nadimi, and S. S. Naseralavi, "New time-based surrogate safety measure to assess crash risk in car-following scenarios," *Transp. Lett.*, vol. 7, no. 4, pp. 229–238, Sep. 2015.
[37] S. Hirst and R. Graham, "The format and perception of collision warnings," in *Ergonomics and Safety of Intelligent Driver Interfaces*, Y. I. Noy, Ed. Mahwah, NJ, USA: Lawrence Erlbaum Associates, 1997, pp. 203–219.
[38] K. Ismail, T. Sayed, and N. Saunier, "Automated analysis of pedestrian–vehicle conflicts: Context for before-and-after studies," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2198, no. 1, pp. 52–64, Jan. 2010.
[39] F. Bella and R. Russo, "A collision warning system for rear-end collision: A driving simulator study," *Proc. Social Behav. Sci.*, vol. 20, pp. 676–686, Jan. 2011.
[40] T. Sayed, G. Brown, and F. Navin, "Simulation of traffic conflicts at unsignalized intersections with TSC-sim," *Accident Anal. Prevention*, vol. 26, no. 5, pp. 593–607, Oct. 1994.
[41] S. Das and A. K. Maurya, "Defining time-to-collision thresholds by the type of lead vehicle in non-lane-based traffic environments," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 12, pp. 4972–4982, Dec. 2020.
[42] Y. Song, Z. Xie, X. Wang, and Y. Zou, "MS-YOLO: Object detection based on YOLOv5 optimized fusion millimeter-wave radar and machine vision," *IEEE Sensors J.*, vol. 22, no. 15, pp. 15435–15447, Aug. 2022, doi: 10.1109/JSEN.2022.3167251.
[43] C. Chen, B. Liu, S. Wan, P. Qiao, and Q. Pei, "An edge traffic flow detection scheme based on deep learning in an intelligent transportation system," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 3, pp. 1840–1852, Mar. 2021, doi: 10.1109/TITS.2020.3025687.
[44] A. Pramanik, S. K. Pal, J. Maiti, and P. Mitra, "Granulated RCNN and multi-class deep SORT for multi-object detection and tracking," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 6, no. 1, pp. 171–181, Feb. 2022.
[45] *Next Generation Simulation*, U. S. Department of Transportation, Washington, DC, USA, 2008. [Online]. Available: www.ngsim.fhwa.dot.gov
[46] M. Goldhammer, S. Kohler, S. Zernetsch, K. Doll, B. Sick, and K. Dietmayer, "Intentions of vulnerable road users detection and forecasting by means of machine learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 7, pp. 3035–3045, Jul. 2020.
[47] M. Khakzar, A. Rakotonirainy, A. Bond, and S. G. Dehkordi, "A dual learning model for vehicle trajectory prediction," *IEEE Access*, vol. 8, pp. 21897–21908, 2020.
[48] N. Deo and M. M. Trivedi, "Multi-modal trajectory prediction of surrounding vehicles with maneuver based LSTMs," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, Art. no. 11791184.
[49] S. Dai, L. Li, and Z. Li, "Modeling vehicle interactions via modified LSTM models for trajectory prediction," *IEEE Access*, vol. 7, pp. 38287–38296, 2019.
[50] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi, "Non-local social pooling for vehicle trajectory prediction," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 975–980.
[51] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," 2018, *arXiv:1805.06771*.

**JOSEPH ANTONY** is currently pursuing the Ph.D. degree with the School of Electronics, Vellore Institute of Technology at Chennai Campus, Chennai. He has been associated with Automotive OEMs, for more than two decades. He has presented few papers in international journals and conferences. His research interests include ADAS, deep learning, and machine vision. He is a Life Member of ISTE.

**SUCHETHA M** is currently a Professor and the Deputy Director, Centre for Healthcare Advancement, Innovation and Research, Vellore Institute of Technology, Chennai campus, Chennai. She has presented papers in international journals and conferences. Her research interests include bio-signal processing using wavelet transform, EMD, adaptive filters, analysis of various filtering methods, non-invasive breath analysis for diabetes detection, sleep apnea detection, and capnographic sensing. She is a Life Member of ISTE.