Trajectory Reconstruction using Probabilistic Time-Space Diagram
CEE 497 Indepedent Study Report

Louis Sungwoo Cho
Advisor: Professor Alireza Talebpour

Department of Civil and Environmental Engineering at
University of Illinois at Urbana-Champaign

May 10th, 2024

Ensuring the integrity of vehicle trajectory data is essential to guarantee the reliability of any traffic flow analyses. Regardless of the data collection method (vehicle-based, infrastructure-based, and aerial) or the sensors utilized for the data collection, gaps are expected in the trajectory data (e.g., overpasses in aerial videography or trees in infrastructure-based approach can block sensors' field of view and result in gaps). Accordingly, trajectory reconstruction and filling the gaps in vehicle trajectory data is an essential step in this process. The majority of existing datasets have utilized a combination of manual and automated approaches to address this problem. Such an approach is significantly time-consuming, preventing the collection and sharing of large-scale vehicle trajectory datasets. Accordingly, this study utilizes the concept of probabilistic time-space diagrams to address the challenge of reconstructing and automating the extraction of vehicular trajectory data. Leveraging a Long Short-Term Memory (LSTM) network based Convolutional Social Pooling (CSP) model, the proposed methodology generates potential trajectories for various vehicle maneuvers, evaluating them against the surrounding environment to determine the most suitable path based on statistical parameters derived from a probabilistic distribution of longitudinal and lateral coordinates generated by the CSP model. Each set of possible trajectories is analyzed with the respective probability distribution by calculating the line integral, and the trajectory with the maximum line integral is selected as the predicted trajectory. The findings suggest that the model is capable of filling the gaps in the vehicle trajectory data. Moreover, the value of the line integral can treated as a confidence measure to identify the need for manual adjustments.

Keywords: Trajectory Prediction, Convolutional Social Pooling Model, LSTM, Safety, Probabilistic Distribution, Low Visibility, Vehicle Maneuvers, Line Integral

2 Introduction

Since the introduction of the Next Generation Simulation (NGSIM) dataset [1], vehicle trajectory data has become an integral element of traffic flow analyses. Accordingly, various vehicle trajectory datasets have been introduced, including but not limited to HighD [2], pNEUMA [3], Third Generation Simulation (TGSIM) [4], and I-24 MOTION [5]. Depending on the area of coverage, one or multiple cameras (either infrastructure-based and stationary or moving, e.g., drones and helicopters) have been used. Extracting vehicle trajectory data faces certain challenges, including vehicle detection accuracy, vehicle tracking, and trajectory smoothing. While addressing the above challenges is not always straightforward, a considerable body of literature is available for each of the aforementioned challenges. In fact, the quality and accuracy of detection, tracking, and smoothing algorithms have significantly increased over the past decade.

However, another key challenge is that there has not been investigated extensively in the literature and can significantly influence the quality of the vehicle trajectory data (especially as the area of coverage increases beyond a single camera). The presence of missing data points in the vehicle trajectory data can compromise the integrity of the dataset. Missing data points occur due to several factors, including obstacles (e.g., overpasses), detection failures, and, in the case of multi-camera detection, the handover process from one camera to the next. The key challenge is connecting the trajectory points that belong to the same vehicle throughout the data collection site (e.g., before and after an overpass). Possibly, the manual process is the most accurate methodology to address these challenges [4]. However, such a process is time-consuming and prevents the extraction of large-scale vehicle trajectory data.

The primary objective of this study is to enhance the accuracy and reliability of vehicle trajectory data extraction by addressing the missing data points challenge. To address this challenge, the main contribution of this study is to propose a robust automated process to connect extracted vehicle trajectories to form a complete dataset. Consider the case of an overpass, where a vehicle trajectory is observed before reaching an overpass. There are, however, multiple potential trajectories after the overpass that can be considered as the continuation of the observation before the overpass. In the manual method, the vehicles should be compared based on the images collected to identify the correct trajectory among the candidates. This study, however, utilizes the Convolutional Social Pooling (CSP) approach [6] to generate a set of potential trajectories based on the existing observation before the overpass and the corresponding driver behavior. The CSP-generated trajectories are then compared to the set of possible trajectories after the overpass. The best trajectory that can represent the continuation of the observed trajectory before the overpass is selected through an optimization process.

The remainder of this paper is structured as follows: Next section presents an overview of some of the existing trajectory prediction algorithms. The Methodology section details the architecture of the CSP model and the algorithm used for trajectory prediction. The Data section describes the TGSIM dataset utilized for this study and specifies the trajectory data analyzed. In the Results and Discussion section, the accuracy of the CSP model is compared with that of the Kalman Filtering method and the Constant Speed method, highlighting their performance differences. Finally, the Conclusion provides a summary of the findings and suggestions for future research.

3 Background

This section aims to provide a brief overview of existing approaches for vehicle trajectory prediction. It is important to note that this section does not provide a comprehensive review of the literature as this area of research has been explored extensively. A detailed review of the literature in this area is presented by Huang et al. [7] and Paravarzar and Mohammad [8].

The majority of trajectory prediction methods have been developed for motion planning algorithms since considering the surrounding traffic environment and its evolution over time are necessary for safety and collision avoidance in autonomous driving. Vehicular trajectory data, however, have not been extensively utilized for training vehicle trajectory prediction algorithms, despite their potential applicability for this purpose.

The key assumption in many of the studies in this area is that the data on the vehicle's current state (e.g., location, speed, headway, etc.) is sufficient for the motion prediction model to accurately identify the future state of the target vehicle [9, 10]. Several studies have shown that various variations of Kalman Filter can be utilized in motion prediction under uncertainty [11, 12]. Broadhurst et al. [13] showed that all possible future states can be simulated using Monte Carlo Methods with vehicular trajectory data as inputs. In fact, Hosseini and Talebpour [14] showed that if the motion models generate any infeasible future states, an elimination process can be performed to keep the set of possible trajectories in a feasible range. However, physics-based models are strongly not recommended because they do not account for interactions between vehicle maneuvers and are restricted to short-term trajectory prediction [14, 10].

Considering the limitations of the physics-based approaches, various data-driven methods for trajectory prediction have been proposed, such as time-series forecasting [15, 16], linear regression [17, 18], support vector regression [19, 20], and variations of neural networks [21, 22]. Some studies have been able to identify vehicle maneuvers using partially observed trajectory data utilizing various machine learning models, such as support vector machines (SVM) [23], multi-layer perceptrons (MLP) [24], logistic regression [25], and recurrent neural networks (RNN) [26].

The Gaussian Process is widely recognized as an optimal modeling approach for trajectory prototyping [14, 27, 28, 29]. Partially observed trajectories are usually compared to prototype trajectories, with the closest prototype being used to predict the rest of the possible trajectories.

Deep learning methods that consider the driving environment, such as the trajectory history of individual and surrounding vehicles, are common for trajectory prediction [30, 31, 32]. Recurrent neural networks (RNN) [31], convolutional neural networks (CNN) [33, 34], or their combinations are often used in these methods [6, 14]. RNNs and their variations are effective for predicting sequential data, while CNNs excel in image and object detection. Combining these architectures is suitable for environments where both sequential and object data need to be analyzed for predicting vehicle motion.

Considering the numerous advantages of deep learning methods, such as the ability to handle complex data patterns in prediction tasks, this study leverages the Convolutional Social Pooling (CSP) model. The CSP model, initially proposed by Deo and Trivedi [6], has demonstrated substantial efficacy in trajectory prediction tasks by effectively capturing the spatial-temporal interactions between agents. Building on this foundation, this study elaborates on the CSP model proposed by Hosseini and Talebpour [14], which introduces several modifications to further improve prediction performance. These enhancements include optimized pooling mechanisms and advanced feature extraction techniques that contribute to the model's performance. The details of the model's architecture and implementation are presented in the next section.

4 Methodology

Extracting the movement of vehicles on freeways presents a significant challenge, primarily because of the complex and varied maneuvers vehicles can execute. These maneuvers include sudden acceleration or deceleration, uneven travel speed, and unexpected turns, making continuous tracking a complex task. This process becomes even more challenging when vehicles cannot be continuously observed or tracked. One such environment is when vehicles pass underneath overpasses, where the line of sight is often obstructed, leading to a temporary loss of visual contact.

Furthermore, inevitable environmental factors such as lighting conditions, weather variations, and traffic density can further complicate the detection and tracking process. In such scenarios, advanced surveillance systems and algorithms play a crucial role. These systems must be adept at reacquiring lost targets and reconnecting the trajectories of vehicles once they re-emerge from obstructions like overpasses. This re-identification process is not just a matter of tracing the vehicle's path but also involves recognizing and matching the vehicle's unique characteristics, such as vehicle ID, model, color, and license plate, to ensure continuity. This section presents our approach to utilizing driver behavior to automate the re-identification process.

4.a Trajectory Re-Identification Algorithm

This section presents the details of the trajectory re-identification algorithm. The core of this algorithm is the CSP model that offers a probabilistic prediction of future trajectories for a target vehicle considering various longitudinal and lateral maneuvers (a total of six maneuvers are considered by this model - see the next section for more details). The CSP utilizes past observation of the target vehicle's behavior as well as its surrounding vehicles to perform the prediction. A high-level introduction to the CSP model is presented in the next section. Detailed integration of the CSP model into individual vehicle traffic state prediction processes can be referred to in the work of Hosseini and Talebpour [14].

Utilizing the trajectory data before the overpass, for each maneuver, the CSP model produced a conditional probability distribution for each time step (0.2s in this study) during the prediction horizon. The key to the proposed algorithms is to identify whether this conditional probability distribution matches the correct trajectory after the overpass. To achieve this, given a set of possible trajectories after the overpass, the line integral value of each trajectory is calculated with respect to the conditional probability distribution. In this study, both the possible set of trajectories and the probabilistic distribution were aligned to a time range of 5s from the beginning of the overpass location. The line integral values were stored with each possible trajectory, respectively. The trajectory with the highest line integral value was then selected as the predicted trajectory for the target vehicle. Figure 1 illustrates the flow chart for the trajectory re-identification process. The underlying algorithm for predicting optimal trajectories is also presented in Algorithm 1.

It is important to emphasize that the CSP mechanism is critical for accurately capturing the interactions between the target vehicle and its surrounding vehicles. This mechanism aggregates spatial and temporal information from neighboring vehicles, allowing the model to understand the dynamics and dependencies within the traffic environment. By doing so, the model can generate more realistic and context-aware trajectory predictions, especially in scenarios where vehicle interactions are complex and highly dynamic [6, 14].

To ensure accurate trajectory prediction and prevent misfitting, the probability distributions for the longitudinal and lateral coordinates (i.e., μ_X and μ_Y) were scaled using a standard min-max scaler. Additionally, a threshold gradient value of 12 was assigned to α to properly align the distribution with the possible trajectories. This scaling and alignment process ensures that the generated trajectories are consistent with the observed vehicle movements. The scaling process also allowed any noise data to be removed and enhanced the accuracy of the predictions. The scaling process is presented in Algorithm 2.

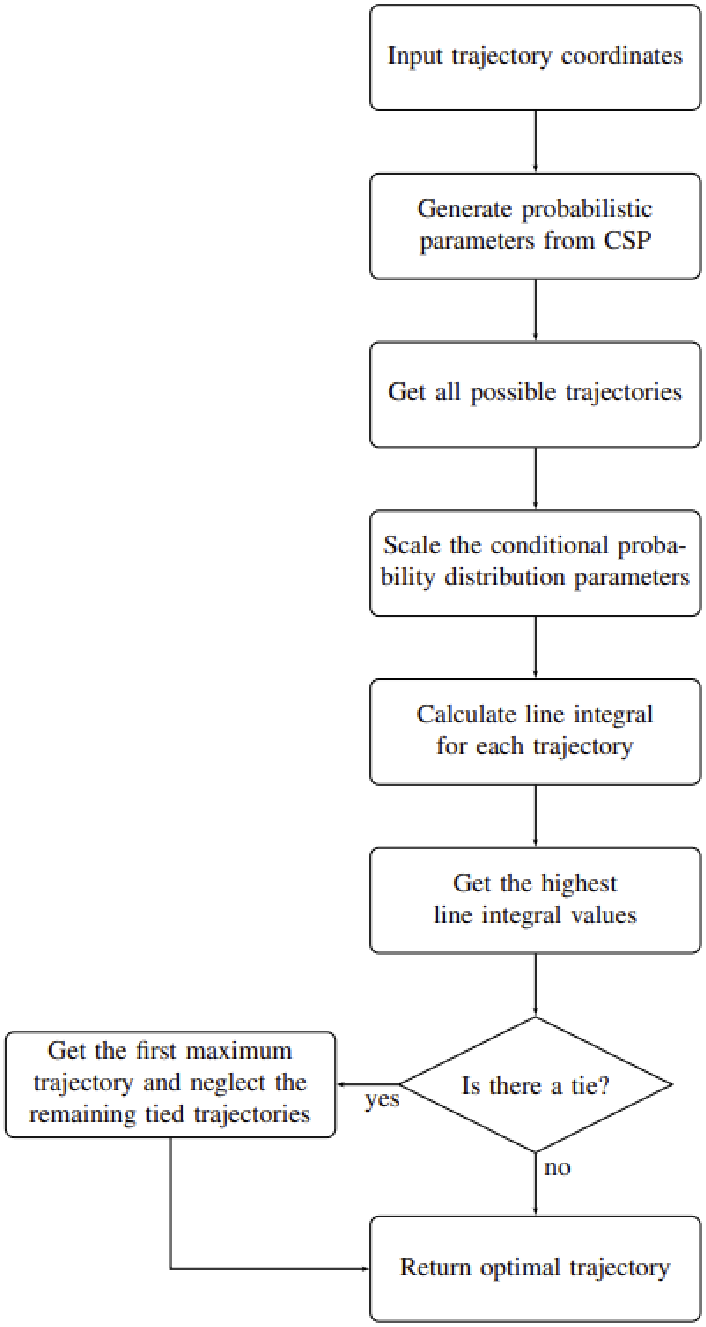


Figure 1: Workflow of the CSP trajectory prediction process.

Algorithm 1 Predict Optimal Trajectories

Input: Trajectory coordinates $x = [x_1, x_2, ..., x_n], y = [y_1, y_2, ..., y_n]$
Output: Optimal Trajectory $x^* = [x_1^*, x_2^*, ..., x_n^*], y^* = [y_1^*, y_2^*, ..., y_n^*]$
Init empty variable for best trajectory*
Init highest line integral value
for all $predictions \in maneuvers$ **do**
    $\mu_X, \mu_Y, \sigma_X, \sigma_Y \leftarrow$ parameters from *predictions*
    scale $(\mu_X, \mu_Y, \sigma_X, \sigma_Y)$
    $cost \leftarrow 0$
    for all $x_i, y_i \in trajectories$ **do**
        $integral_i \leftarrow$ call *LineIntegral* with x_i, y_i, x_{i+1}, y_{i+1}, μ_{X_i}, μ_{Y_i}, σ_{X_i}, σ_{Y_i}
        $integral_{i+1} \leftarrow$ call *LineIntegral* with x_{i+1}, y_{i+1}, x_{i+2}, y_{i+2}, $\mu_{X_{i+1}}$, $\mu_{Y_{i+1}}$, $\sigma_{X_{i+1}}$, $\sigma_{Y_{i+1}}$
        $cost \leftarrow cost + integral_i + integral_{i+1}$
    end for
    if $cost > value_{max}$ **then**
        $value_{max} \leftarrow cost$
        $temp_x \leftarrow x$
        $temp_y \leftarrow y$
        $trajectory* \leftarrow$ update with $temp_x$, $temp_y$
    end if
end for
return $trajectory*$

Algorithm 2 Scale Distribution

Input: $x_{\text{list}}, \mu_{X_{\text{before}}}, \mu_{Y_{\text{before}}}$, overpass_start_loc_x, overpass_length, overpass_start_loc_y, α
Output: μ_X, μ_Y
gradient $\leftarrow \max(x_{\text{list}}) - \min(x_{\text{list}})$
if gradient $\leq \alpha$ **then**
    gradient $\leftarrow$ gradient $+ \alpha$
end if
$\mu_{X_{\text{scaled}}}, \mu_{Y_{\text{scaled}}} \leftarrow$ scale_data($\mu_{X_{\text{before}}}, \mu_{Y_{\text{before}}}$, method='minmax')
$\mu_X \leftarrow [(\text{gradient} \times m_x) + \text{overpass\_start\_loc\_x} + \text{overpass\_length} \mid m_x \in \mu_{X_{\text{scaled}}}]$
$\mu_Y \leftarrow [m_y + \text{overpass\_start\_loc\_y} \mid m_y \in \mu_{Y_{\text{scaled}}}]$

4.b Convolutional Social Pooling

As discussed above, the CSP model is at the core of the presented approach. This model, developed by Deo and Trivedi [6], utilizes an encoder-decoder structure and an LSTM structure to predict the probability distribution of a target vehicle's future positions. This model calculates the parameters of the conditional distribution based on the vehicle's observed trajectory histories. It differentiates between various maneuvers by considering both longitudinal and lateral movements. The longitudinal movements encompass braking and not braking, while the lateral movements include maintaining the current lane or changing lanes to the right or left.

This model was trained to estimate the likelihood of different maneuver combinations and predict the probabilistic future location of the target vehicle. For each time step in the future and for each maneuver, the model provides the mean and covariance of a bivariate normal distribution. Essentially, it predicts a unique set of parameters for every future time step and maneuver combination [6]. The detailed model architecture is shown in Figure 2. Figure 3 also illustrates the range of potential paths, highlighting the effectiveness of the probabilistic approach in capturing the variability and uncertainty inherent in vehicle movements. More details about the structure of this model can be found in Hosseini and Talebpour [14].

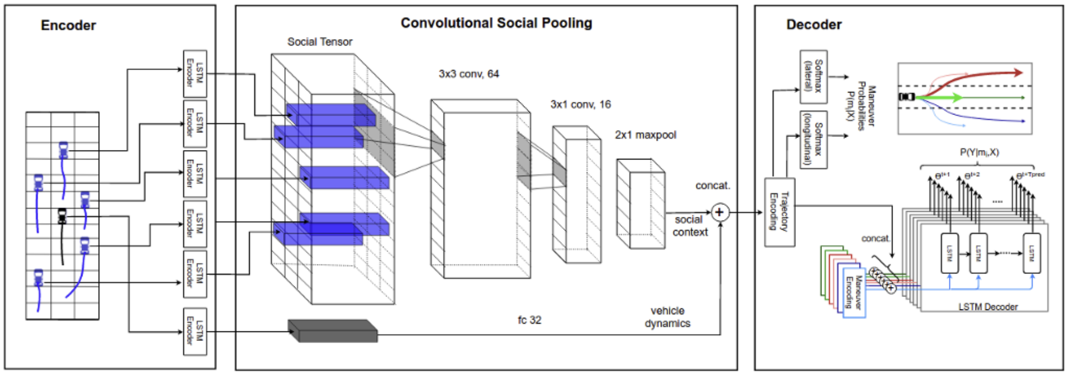


Figure 2: The encoder, an LSTM with shared weights, captures vehicle dynamics from trajectory histories. Convolutional Social Pooling layers are used to understand the spatial relationships between different tracks. The maneuver-based decoder then generates a multi-modal predictive distribution for the future motion of the target vehicle from Deo and Trivedi [6], Convolutional Social Pooling for Vehicle Trajectory Prediction, Figure 3.

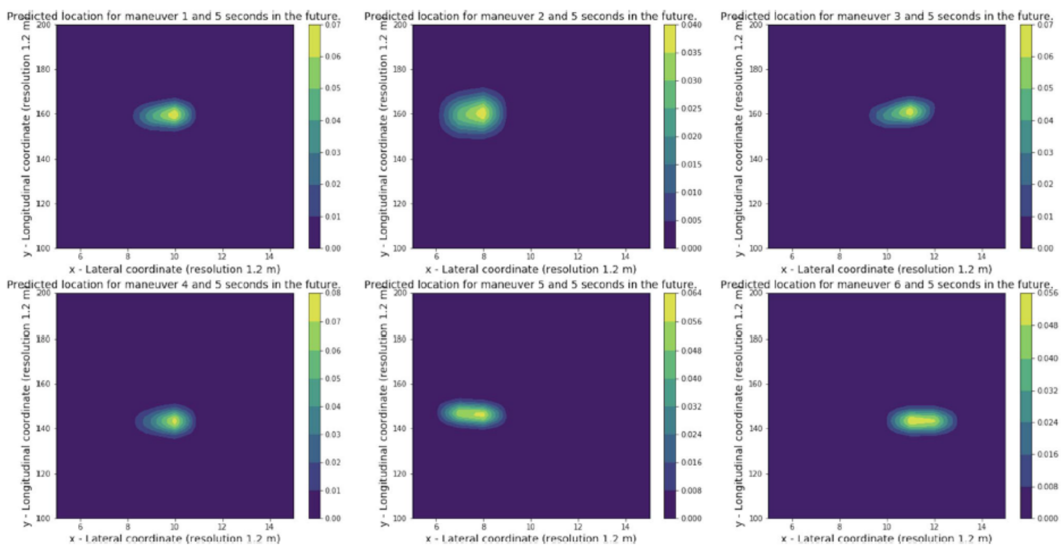


Figure 3: Generated plots showing the predicted location of a single vehicle for 5 seconds in the future for each maneuver from Hosseini, Talebpour, 2023, Probabilistic Traffic State Prediction Based on Vehicle Trajectory Data, Figure 3. [14].

4.c Matching Cost

The matching between the predicted conditional probability distributions for each maneuver and the set of possible trajectories is determined based on the line integral. The approach is designed to compute the line integral of a distribution across a path in a two-dimensional space, defined by the start and end trajectory coordinates. Accordingly, at time i, the coordinates can be represented as (x_{i-1}, y_{i-1}) and (x_{i+1}, y_{i+1}). The line integral formula was defined by these points and a set of Gaussian distributions (i.e., CSP predictions) characterized by parameters μ_x, μ_y, and σ. The x and y coordinates of the trajectories were incremented by 0.1s time step. The Gaussian parameters were incremented by 0.2s time steps. Due to the different time stamps of the x and y coordinates and μ_x, μ_y, and σ, the x and y coordinates were divided into two segments. The line integral cost was computed using the following equations:

$$\text{cost} = \sum \left(\frac{e^{\left(\frac{b^2}{4a} - c\right)}}{2\pi\sigma} \cdot \frac{1}{\sqrt{a}} \cdot \sqrt{\pi} \cdot \frac{1}{2} \cdot \left(\text{erf}\left(\sqrt{a} + \frac{b}{2\sqrt{a}}\right) - \text{erf}\left(\frac{b}{2\sqrt{a}}\right) \right) \cdot \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \right) \tag{1}$$

$$\epsilon = 8 \times 10^{-5} \tag{2}$$

$$\sigma = \sqrt{\frac{\sigma_X^2 + \sigma_Y^2}{2}} + \epsilon \tag{3}$$

$$a = \left((x_1 - x_2)^2 + (y_1 - y_2)^2\right) \left(\frac{1}{2\sigma}\right) + \epsilon \tag{4}$$

$$b = \left(-2x_1^2 + 2x_1x_2 + 2x_1\mu_X - 2x_2\mu_X + -2y_1^2 + 2y_1y_2 + 2y_1\mu_Y - 2y_2\mu_Y\right) \left(\frac{1}{2\sigma}\right) \tag{5}$$

$$c = \left((x_1 - \mu_X)^2 + (y_1 - \mu_Y)^2\right) \left(\frac{1}{2\sigma}\right) \tag{6}$$

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt \tag{7}$$

The integral was computed along the path between (x_{i-1}, y_{i-1}) and (x_i, y_i), and between (x_i, y_i) and (x_{i+1}, y_{i+1}) incorporating the influence of Gaussian distributions along this path where i is the index of the trajectory points. Furthermore, to avoid noise and to scale the data accordingly to the trajectories, the μ_X and μ_Y values were normalized using the min-max scaler function. The Min-Max scaling formula can be given by:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{8}$$

This method effectively integrates the influence of a set of Gaussian-like functions along the straight line connecting the given points. The trajectory with the highest line integral value is chosen as the projected trajectory for the vehicle passing under the overpass. The use of Gaussian-like functions allows for the modeling of uncertainties and variations in vehicle trajectories, capturing the probabilistic nature of real-world traffic flow. By considering the distribution of possible trajectories rather than a single deterministic path, this method can better accommodate unexpected changes in vehicle movement and traffic conditions. If a tie between the maximum line integral values were to exist, the maximum value of the first distribution would be chosen and the remaining would be neglected. This approach ensures stability by avoiding the selection of ambiguous trajectories or probabilistic distribution that could result from noise or minor variations in the vehicular trajectory data.

The effectiveness of this method lies in its probabilistic foundation, which contrasts with traditional deterministic methods that predict a single, fixed trajectory. Deterministic methods often fail to capture the inherent uncertainties and dynamic variations present in real-world traffic environments. By incorporating Gaussian-like functions, this method provides a more flexible and adaptive prediction model, capable of handling the complexities of freeway traffic.

5 Data Description

The primary data source for vehicle trajectory analysis in this study is the Third Generation Simulation (TGSIM) dataset[35, 4]. The TGSIM dataset is particularly valuable due to its focus on complex urban multi-modal roadway networks, which significantly contributes to research in the field of Connected and Automated Vehicles (CAV) and the development of efficient advanced vehicular systems [4].

Data from Interstate 294 was utilized in this study to train and evaluate a probabilistic trajectory prediction model. The TGSIM dataset, specifically designed to capture high-fidelity vehicular movements, includes data collected at a frequency of 10 Hz [4]. Each data point within a vehicle's trajectory encompasses several features: time measured in seconds, longitudinal (x) and lateral (y) coordinates measured in meters, velocity measured in meters per second, and acceleration measured in meters per second squared. This comprehensive dataset enables a detailed analysis of vehicular movement patterns.

A descriptive analysis of the dataset revealed that it effectively captures a range of traffic conditions, from free-flow to congested states, providing insights into speed and acceleration distributions [35]. The sufficient data ensures reliable modeling and validation of traffic scenarios, particularly those of interest in this study, such as under highway overpasses. Figure 4 shows the trajectories for each vehicle, with both incoming and outgoing trajectories plotted. The segment between 1800 meters and 1815 meters indicates a sample overpass location where trajectory data were missing. The overpass is assumed to be 15 meters (50 ft) wide.

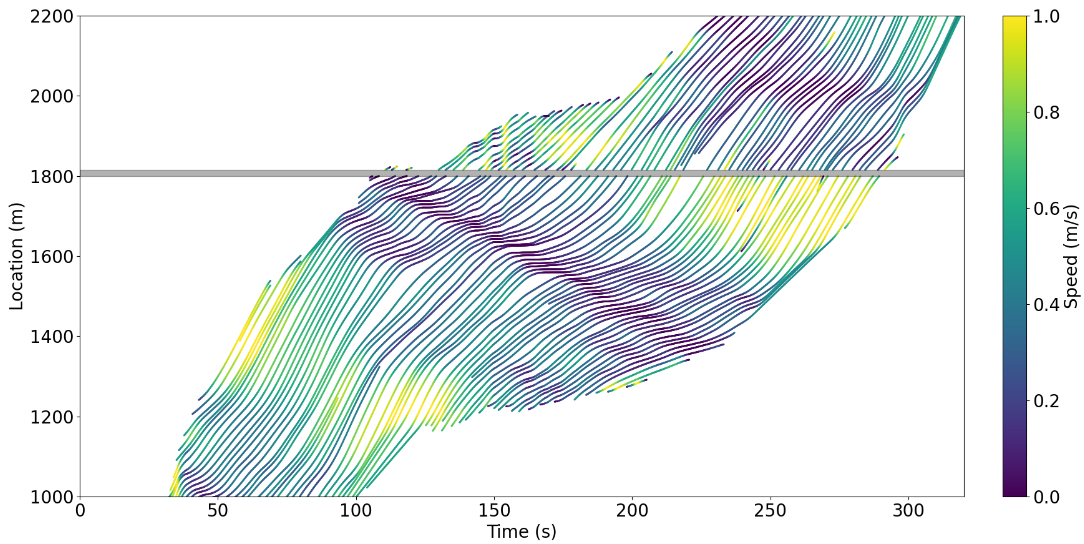


Figure 4: The time-space diagram for the trajectory data of each vehicle. The horizontally highlighted grey bar indicates the location of the 50 ft (15 meters) wide overpass where data is missing. Vehicle speeds are represented using a color gradient: higher speeds are shown in lighter colors, while lower speeds are depicted in darker colors.

6 Results and Discussion

The CSP model is trained and evaluated using the TGSIM dataset. This dataset provides trajectory data with time step increments of 0.1 seconds for both longitudinal and lateral positions of each vehicle. The dataset is divided into training (80%), validation (5%), and testing (15%) sets. The trajectory prediction model developed by Deo and Trivedi [6] is trained on the training data for each vehicle with a prediction horizon of 5s. After training, the model calculated the conditional probability distribution parameters of the possible vehicle positions in both the longitudinal and lateral directions. Additionally, the probability of each maneuver is generated with time steps of 0.2s over the next 5s for trajectory prediction using the lateral and longitudinal coordinates (following the same process as Hosseini and Talebpour [14]).

Using the conditional probability distributions generated by the CSP model, a set of possible trajectories for each vehicle is selected within the overpass location. The probabilistic distribution parameters of μ_X, μ_Y, σ_X, and σ_Y are used to perform the line integral calculation for each possible trajectory set of the target vehicle. The trajectory with the highest line integral value is then chosen as the predicted trajectory. This predicted trajectory is then compared to the actual trajectory to compute accuracy as a percentage of correctly identified trajectories.

The results indicate that the proposed model using the line integral method can achieve an accuracy rate between 70.00% and 80.00% in different overpass locations as shown in Table 2. Most of the trajectories are predicted accurately for the target vehicles. The performance of the proposed approach is also compared to the Constant Speed and Kalman Filter based prediction approaches. In contrast, the Constant Speed method, which computes the error between the generated straight-line plot and the set of possible trajectories, achieved an accuracy rate between 16.00% and 35.00%, which is significantly lower than that of the CSP-based model. The accuracy rate for the Kalman Filter based method has fluctuated between 7.00 % to 45.00%. Although in some cases, the Kalman Filtering method has performed better than the Constant Speed method (which is expected since the Kalman Filter based method considers both speed and acceleration), it did not perform as well as the proposed approach in this study.

These findings demonstrate the effectiveness of the CSP model combined with the line integral method for trajectory prediction in complex driving scenarios. It is important to note that as the prediction horizon increases, the accuracy rate decreases (see Table 1). The accuracy rate is around 80% to 96% for a prediction horizon of between 5 to 15 meters, and it is sharply decreased beyond the length of 15 meters. In addition to this, both the Constant Speed method and the Kalman Filter based methods face the same challenge as the prediction horizon increases. The results from Table 1 show that a longer prediction horizon can cause difficulties with extended prediction. Interestingly enough, the prediction of the proposed approach is comparable to the Constant Speed and Kalman Filter based methods when the prediction horizon is 30 meters or longer.

Table 1: Results for CSP, Constant Speed and Kalman Filtering with Fixed Overpass Starting Point of 1800 meters and Different Overpass Lengths.

Overpass Start (m)	Overpass End (m)	CSP	Constant Speed	Kalman Filter
1800	1805	96.12%	35.22%	36.67%
1800	1810	89.73%	27.65%	35.59%
1800	1815	80.66%	29.39%	26.04%
1800	1820	50.75%	21.04%	18.11%
1800	1825	33.00%	17.05%	20.12%
1800	1830	19.75%	16.69%	22.80%

The results of this study demonstrate the effectiveness of the CSP model combined with the line integral method in predicting vehicle trajectories under complex driving scenarios with low visibility. To further assess the performance of the proposed method, different locations for the overpass is also explored. As shown in Table 2, the CSP model achieved an accuracy rate ranging from 70.00% to 80.00% when predicting vehicle trajectories. This significantly outperforms the traditional Constant Speed method, which achieved an accuracy rate between 11.00% and 38.00%, and the Kalman Filter method which achieved an accuracy rate between 7.00% to 45.00%. The higher accuracy of the CSP model highlights its ability to capture the dynamic and uncertain nature of vehicle movements, especially in constrained environments like freeway overpasses.

Table 2: Results for CSP, Constant Speed and Kalman Filtering Methods with Different Overpass Locations and Fixed Overpass Lengths.

Overpass Start (m)	Overpass End (m)	CSP	Constant Speed	Kalman Filter
1800	1815	80.66%	29.39%	26.04%
1895	1910	78.51%	13.51%	17.00%
1165	1180	77.67%	26.77%	28.57%
1120	1135	77.03%	14.65%	45.45%
1930	1945	74.37%	11.76%	20.75%
1050	1065	71.67%	20.24%	7.14%
2111	2126	71.65%	38.06%	35.00%
1705	1720	70.96%	37.00%	22.27%
2065	2080	70.02%	27.75%	24.39%
1755	1770	70.00%	23.75%	28.30%

The performance of the CSP model was significantly enhanced by its ability to process sequential vehicle trajectory data [6]. By incorporating the line integral method, the model evaluates potential trajectories within a probabilistic framework, selecting the most likely path based on the accumulated probabilistic distribution parameters over time [14]. This approach effectively captures the complex patterns of vehicular dynamics, which are often neglected by simpler deterministic methods. As illustrated in Figure 5, the predicted probability distribution generated by the CSP model aligns closely with one of the possible trajectories that the vehicle can take. This alignment underscores the model's precision and its ability to predict realistic and accurate vehicle paths under varying traffic conditions.

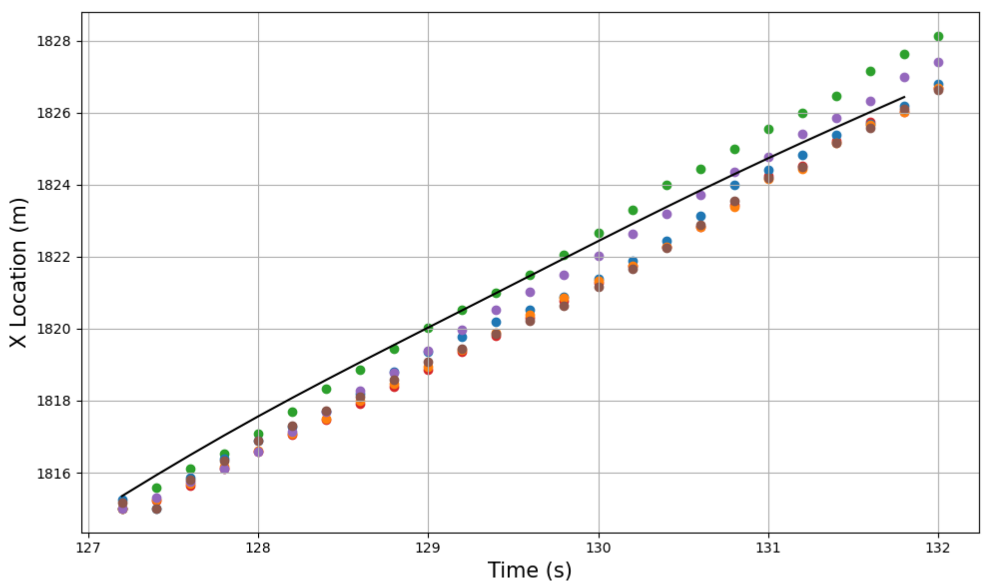


Figure 5: In this example, the overpass was assumed to be between 1800 and 1815 meters. The plotted dots which represent multiple predicted trajectories generated by the CSP model align appropriately for one of the chosen possible vehicular trajectory for the target vehicle. The overpass location between 1800 m to 1815 m can be refered to Figure 4.

It is important to note that the CSP Model does not perform well in areas where the traffic shockwaves start to form under the overpass (see Figures 6 and 7). Further research (and potentially additional calibration) is required to rectify this issue as well as to allow long-range trajectory predictions.

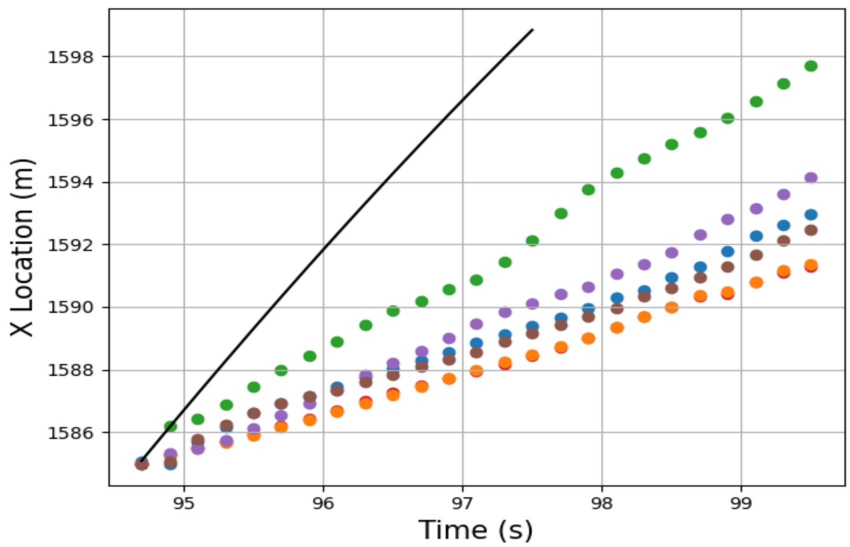


Figure 6: In this example, the overpass was assumed to be between 1570 and 1585 meters. However, the generated predicted trajectories do not align appropriately with one of the chosen possible vehicular trajectory for the target vehicle in this example.

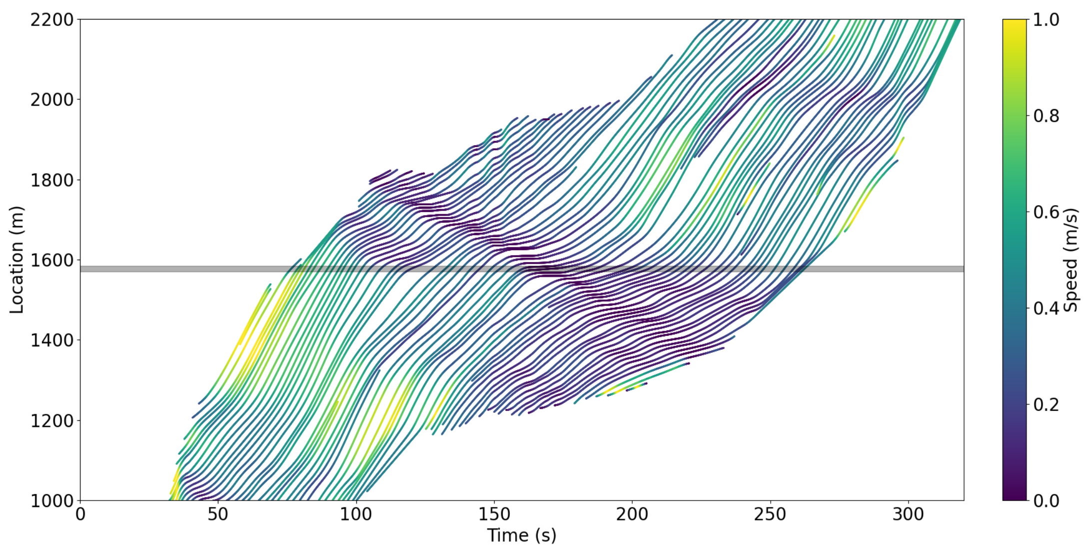


Figure 7: The overpass location 1570 m to 1585 m from Figure 6 can be shown in this trajectory diagram highlighted in grey. Vehicle speeds are represented using a color gradient: higher speeds are shown in lighter colors, while lower speeds are depicted in darker colors.

The distribution of the line integral values is shown in Figure 8. The density is skewed to the right for both the correct cases and incorrect cases for the trajectory prediction process. Specifically, line integral values between 0.0 and 5.0 has a higher density than other ranges, suggesting that most of the predicted trajectories, whether correct or incorrect, fall within this range. This is particularly due to the high density of the vehicle trajectory data where multiple possible trajectories are identified and can be mitigated through further training of the CSP model.

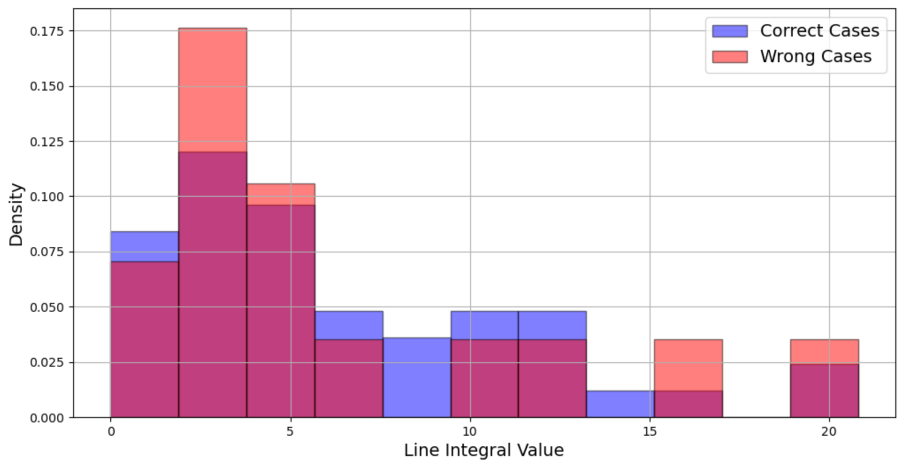
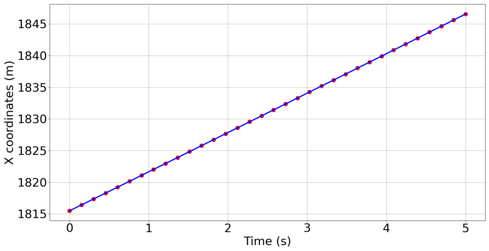


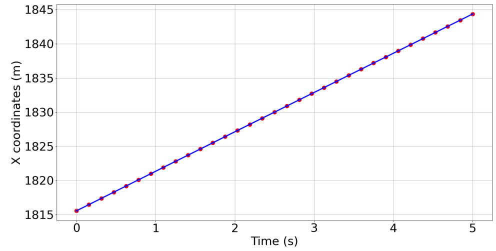
Figure 8: The normal distribution of the line integral values calculated by the prediction function of the correctly and incorrectly predicted trajectories.

However, there were instances where the probability distribution generated by the CSP model did not align well with the actual trajectories. As shown in Figure 6, the predicted trajectories deviated from the actual vehicle paths, highlighting potential areas for improvement. These discrepancies may arise from sudden changes in vehicle behavior, uncertain environmental factors, or limitations in the training data. Addressing these issues could involve incorporating additional contextual information, improving the model's architecture, and enhancing the quality and diversity of the training data to improve the accuracy of the trajectory predictions.

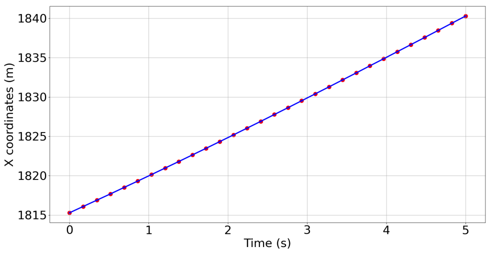
Figure 9 illustrates additional scenarios where the predicted trajectories generated by the CSP model align closely with the actual vehicle paths, demonstrating the model's precision in such cases. This alignment resembles the model's capability to handle real-world complexities and make accurate predictions even in challenging scenarios. In contrast, Figure 10 shows additional scenarios where the predicted trajectories do not align well with the actual paths, indicating potential areas for improvement. Such discrepancies may arise from sudden changes in vehicle behavior, uncertain environmental factors, or limitations in the training data.



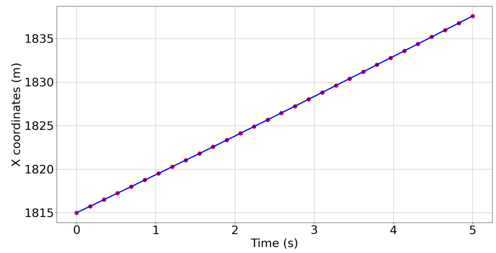
(a)



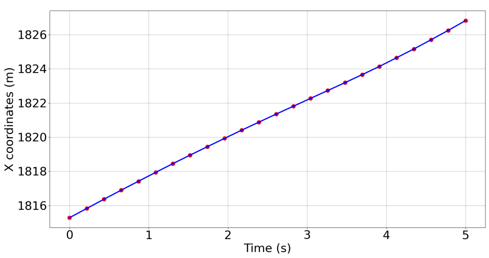
(b)



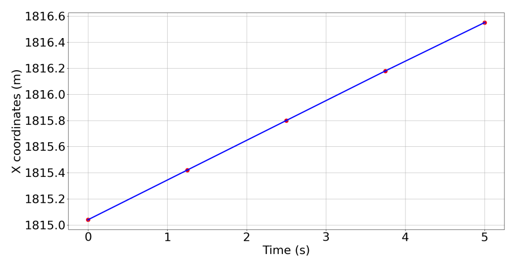
(c)



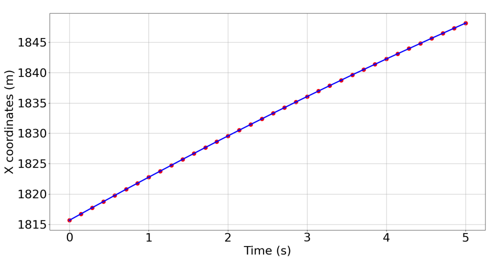
(d)



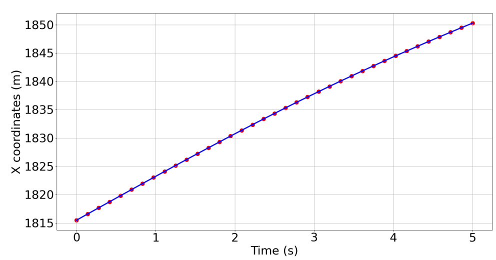
(e)



(f)



(g)



(h)

Figure 9: Trajectory diagrams of 9a, 9b, 9c, 9d, 9e, 9f, 9g, and 9h show the predicted trajectory versus the actual ground truth trajectory for each of the selected target vehicles using the CSP line integral method. Predicted trajectories were marked in red and actual ground truth trajectories were marked in blue.

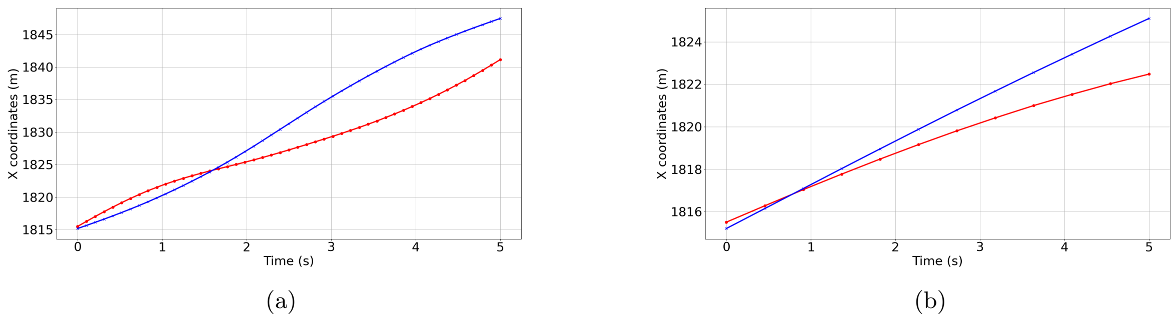


Figure 10: Trajectory diagrams of 10a and 10b show the predicted trajectory versus the actual ground truth trajectory for each of the selected target vehicles using the CSP line integral method. Predicted trajectories were marked in red and actual ground truth trajectories were marked in blue. This case is when the predicted trajectory does not align appropriately with the actual ground truth trajectory.

The presented results highlight the importance of incorporating environmental context into trajectory prediction models. The success of the CSP model in scenarios with clear, predictable vehicle behavior suggests that further enhancements could be achieved by integrating additional contextual information, such as real-time traffic data and sensor inputs. This would likely improve the model's robustness and adaptability to sudden changes in the driving environment.

In addition, the discrepancies observed in Figures 5 and 6 emphasize the need for ongoing refinement of the prediction algorithms. Future work could focus on improving the training data quality and diversity, incorporating more sophisticated modeling techniques to handle the variability in vehicle behaviors, and exploring hybrid models that combine deterministic and probabilistic approaches for enhanced prediction accuracy.

7 Conclusion

This study presents a novel methodology for re-identification of vehicle trajectory data when dealing with missing data points. The presented approach relies on a probabilistic prediction of future vehicle locations (calculated based on the observed target trajectory) to identify the most likely candidate as the continuation of the target trajectory. The results of this study highlight the significant potential of the proposed model in accurately reconstructing vehicle trajectories under complex driving scenarios. The proposed model demonstrated a high accuracy rate ranging from 70.00% to 80.00%, significantly outperforming the traditional Constant Speed based method, which achieved an accuracy rate between 11.00% and 38.00%, and the Kalman Filter based method which achieved an accuracy rate between 7.00% to 45.00%. This performance underscores the model's ability to capture the dynamic and uncertain nature of vehicle movements, particularly in constrained environments such as freeway overpasses.

The proposed method also faces certain limitations, including loss of accuracy as the prediction horizon increases. This suggests a need for model calibration and enhancement to improve long-range trajectory predictions. Future research will focus on incorporating additional information, such as real-time traffic data and sensor inputs, to enhance the model's adaptability to sudden changes in the driving environment. Improving the quality and diversity of training data, refining the model's architecture, and exploring hybrid models that combine deterministic and probabilistic approaches are crucial steps for advancing the model's accuracy.

References

[1] Vincenzo Punzo, Maria Teresa Borzacchiello, and Biagio Ciuffo. On the assessment of vehicle trajectory data accuracy and application to the next generation simulation (ngsim) program data. *Transportation Research Part C: Emerging Technologies*, 19(6):1243–1262, 2011.

[2] Robert Krajewski, Julian Bock, Lennart Kloeker, and Lutz Eckstein. The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 2118–2125. IEEE, 2018.

[3] Emmanouil Barmpounakis and Nikolas Geroliminis. On the new era of urban traffic monitoring with massive drone data: The pneuma large-scale field experiment. *Transportation Research Part C: Emerging Technologies*, 111:50–71, 2020.

[4] Rami Ammourah, Pedram Beigi, Bingyi Fan, Samer H. Hamdar, John Hourdos, Chun-Chien Hsiao, Rachel James, Mohammadreza Khajeh-Hosseini, Hani S. Mahmassani, Dana Monzer, Tina Radvand, Alireza Talebpour, Mahdi Yousefi, and Yanlin Zhang. Introduction to the third generation simulation dataset: Data collection and trajectory extraction. *Transportation Research Record*, pages 1–17, 2024. doi: 10.1177/0361198124125757.

[5] Derek Gloudemans, Yanbing Wang, Junyi Ji, Gergely Zachár, William Barbour, Eric Hall, Meredith Cebelak, Lee Smith, and Daniel B. Work. I-24 motion: An instrument for freeway traffic science. *Transportation Research Part C: Emerging Technologies*, 155:104311, 2023. ISSN 0968-090X. doi: https://doi.org/10.1016/j.trc.2023.104311. URL `https://www.sciencedirect.com/science/article/pii/S0968090X23003005`.

[6] Nishit Deo and Mohan M. Trivedi. Convolutional social pooling for vehicle trajectory prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1468–1476, 2018. Accessed on 2024-07-22.

[7] Renbo Huang, Guirong Zhuo, Lu Xiong, Shouyi Lu, and Wei Tian. A review of deep learning-based vehicle motion prediction for autonomous driving. *Sustainability*, 15(20):14716, 2023.

[8] Shahrokh Paravarzar and Belqes Mohammad. Motion prediction on self-driving cars: A review. *arXiv preprint arXiv:2011.03635*, 2020.

[9] M. Khajeh Hosseini, A. Talebpour, and S. Shakkottai. Privacy risk of connected vehicles in relation to vehicle tracking when transmitting basic safety message type 1 data. *Transportation Research Record*, 2673(12):636–643, 2019.

[10] Stéphane Lefèvre, David Vasquez, and Christian Laugier. A survey on motion prediction and risk assessment for intelligent vehicles. *ROBOMECH Journal*, 1(1):1–14, 2014.

[11] T. Batz, K. Watson, and J. Beyerer. Recognition of dangerous situations within a cooperative group of vehicles. In *2009 IEEE Intelligent Vehicles Symposium*, pages 907–912. IEEE, 2009.

[12] M. T. Abbas, M. A. Jibran, M. Afaq, and W.-C. Song. An adaptive approach to vehicle trajectory prediction using multimodel kalman filter. *Transactions on Emerging Telecommunications Technologies*, 31(5):e3734, 2020.

[13] A. Broadhurst, S. Baker, and T. Kanade. Monte carlo road safety reasoning. In *IEEE Proceedings. Intelligent Vehicles Symposium, 2005*, pages 319–324. IEEE, 2005.

[14] Mohammadreza Khajeh Hosseini and Alireza Talebpour. Probabilistic traffic state prediction based on vehicle trajectory data. *SpringerLink*, 2023.

[15] C. Chen, J. Hu, Q. Meng, and Y. Zhang. Short-time traffic flow prediction with arima-garch model. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 607–612, 2011.

[16] S. V. Kumar and L. Vanajakshi. Short-term traffic flow prediction using seasonal arima model with limited input data. *European Transport Research Review*, 7(3):21, 2015.

[17] B. Smith, W. Scherer, and J. Conklin. Exploring imputation techniques for missing data in transportation management systems. In *Transportation Research Record*, volume 1836, pages 132–142, 2003.

[18] M. R. Wilby, J. J. V. Díaz, A. B. Rodríguez González, and M. Á. Sotelo. Lightweight occupancy estimation on freeways using extended floating car data. *Journal of Intelligent Transportation Systems*, 18(2):149–163, 2014.

[19] H. Su, L. Zhang, and S. Yu. Short-term traffic flow prediction based on incremental support vector regression. In *ICNC Third International Conference on Natural Computation*, volume 1, pages 640–645, 2007.

[20] M. Castro-Neto, Y. S. Jeong, M. K. Jeong, and L. D. Han. Online-svr for short-term traffic flow prediction under typical and atypical traffic conditions. *Expert Systems with Applications*, 36(3):6164–6173, 2009.

[21] K. Y. Chan, T. S. Dillon, J. Singh, and E. Chang. Neural-network-based models for short-term traffic flow forecasting using a hybrid exponential smoothing and levenberg-marquardt algorithm. *IEEE Transactions on Intelligent Transportation Systems*, 13(2):644–654, 2012.

[22] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang. Traffic flow prediction with big data: a deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*, 16(2):865–873, 2015.

[23] Y. Dou, F. Yan, and D. Feng. Lane changing prediction at highway lane drops using support vector machine and artificial neural network classifiers. In *2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 901–906. IEEE, 2016.

[24] S. Yoon and D. Kum. The multilayer perceptron approach to lateral motion prediction of surrounding vehicles for autonomous vehicles. In *2016 IEEE Intelligent Vehicles Symposium (IV)*, pages 1307–1312. IEEE, 2016.

[25] S. Klingelschmitt, M. Platho, H.-M. Groß, V. Willert, and J. Eggert. Combining behavior and situation information for reliably estimating multiple intentions. In *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pages 388–393. IEEE, 2014.

[26] A. Khosroshahi. *Learning, classification and prediction of maneuvers of surround vehicles at intersections using LSTMs*. PhD thesis, University of California, San Diego, 2017.

[27] Sanaz A. Goli, Behrouz H. Far, and Abraham O. Fapojuwo. Vehicle trajectory prediction with gaussian process regression in connected vehicle environment. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 550–555. IEEE, 2018.

[28] Quang Tran and Jonas Firl. Online maneuver recognition and multimodal trajectory prediction for intersection assistance using non-parametric regression. In *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pages 918–923. IEEE, 2014.

[29] Josh Joseph, Finale Doshi-Velez, Albert S. Huang, and Nicholas Roy. A bayesian nonparametric approach to modeling motion patterns. *Autonomous Robots*, 31(4):383–400, 2011.

[30] Saeed Mozaffari, Othman Y. Al-Jarrah, Mehrdad Dianati, Peter Jennings, and Angelos Mouzakitis. Deep learning-based vehicle behavior prediction for autonomous driving applications: A review. *IEEE Transactions on Intelligent Transportation Systems*, 23:33–47, 2020.

[31] Nishit Deo and Mohan M. Trivedi. Multi-modal trajectory prediction of surrounding vehicles with maneuver based lstms. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1179–1184. IEEE, 2018.

[32] Yao Ma, Xinyue Zhu, Shiqi Zhang, Ruigang Yang, Wen Wang, and Dinesh Manocha. Trafficpredict: Trajectory prediction for heterogeneous traffic-agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6120–6127, 2019.

[33] Hang Cui, Vladan Radosavljevic, Fang-Chieh Chou, Tsung-Han Lin, Thi Nguyen, Tsung-Kuan Huang, Jeff Schneider, and Nemanja Djuric. Multimodal trajectory predictions for autonomous driving using deep convolutional networks. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 2090–2096. IEEE, 2019.

[34] Wenjie Luo, Bin Yang, and Raquel Urtasun. Fast and furious: Real time end-to-end 3d detection, tracking and motion forecasting with a single convolutional net. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3569–3577, 2018.

[35] Alireza Talebpour, Hani S. Mahmassani, and Samer H. Hamdar. Third generation simulation data (tgsim): A closer look at the impacts of automated driving systems on human behavior. Technical Report FHWA-JPO-24-133, Federal Highway Administration, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, May 2024. The project CORs were Rachel James and John Hourdos. The JPO Program Manager was Hyungjun Park.