Turn-Aware LSTM Model for Vehicle Trajectory Forecasting

X. Zhou[1] C. Alecsandru[2] S. Bashbaghi[3] Y. Jeong[1] Y. Chen[1]

[1] Department of Engineering, Concordia University, Montreal, Canada
[2] Associate Professor at Concordia University, Montreal, Canada
[3] Senior Data Scientist, Ericsson, Montreal, Canada
email: xingnan.zhou@mail.concordia.ca; ciprian.alecsandru@concordia.ca; saman.bashbaghi@ericsson.com; hzijeong@gmail.com; ye.chen@mail.concordia.ca

Abstract

Accurate trajectory prediction is essential for autonomous driving safety at intersections. Existing deep learning models often overlook turning behaviors, leading to curvature misestimation. This study proposes a Turn-Aware LSTM network that explicitly encodes maneuver intentions—left, right, or straight—using vehicle trajectories extracted from UAV footage via YOLOv8 and DeepSORT. To mitigate tracking noise, a cumulative turning-angle strategy is introduced for robust maneuver classification. Experiments demonstrate that the proposed model significantly improves prediction accuracy for turning maneuvers, reducing Final Displacement Error (FDE) by 15–20% at a 3-second horizon compared to vanilla LSTM and physics-based baselines. The findings validate the integration of maneuver-aware encoding for enhanced intersection-level forecasting in real-time applications

Keywords – Vehicle trajectory prediction, LSTM, Spatiotemporal relationship, Turning behavior, Encoding

1. Introduction

The coexistence of Connected Autonomous Vehicles (CAVs) and Conventional Vehicles (CONVs) on urban roads poses significant challenges to traffic safety and efficiency [1]. One of the key components for addressing these challenges is trajectory forecasting, which enables autonomous vehicles to anticipate potential conflicts and make informed decisions to enhance traffic flow and safety [2]. However, many existing trajectory prediction approaches heavily rely on external data sources, such as GPS signals and detailed road geometry, which inherently limits their scalability and adaptability [3]. Given the complexity of urban traffic networks, a solution that

uses only vehicle trajectory data to infer lane positions and turning behaviors is essential for real-world deployment and effective decision-making.

Previous studies on Vehicle Trajectory Prediction (VTP) have explored various deep learning-based approaches to improve accuracy. Notably, models such as STA-LSTM, which integrates spatial-temporal attention mechanisms, have enhanced the interpretability of vehicle trajectory predictions by incorporating historical trajectory patterns and interactions with neighboring vehicles [4]. Another promising approach is the Graph Attention Network (GAT) combined with LSTM encoders, which encodes motion data and inter-vehicle relationships to generate robust trajectory forecasts [5]. These approaches have advanced the field, but their predictive accuracy remains limited in urban intersections where vehicles frequently execute left turns, right turns, or lane changes. Most models still rely on position-only trajectory data without explicitly encoding lane-level context or maneuver intentions, leading to degraded performance precisely when turning behavior drives safety-critical outcomes.

In mixed-traffic environments of connected/autonomous vehicles (CAVs) and conventional vehicles (ConVs), accurate forecasting becomes even more important. Autonomous vehicles must anticipate turning and lane-changing maneuvers at intersections and freeway exits to avoid conflicts, yet conventional LSTM-based models—often trained on datasets such as NGSIM—are constrained by fixed-camera viewpoints, occlusion, and limited spatial resolution. These shortcomings reduce their ability to learn lane-specific functions and robustly recognize turning intent, weakening their practical applicability in real-world traffic management [5].

To address these limitations, this study proposes a Turn-Aware LSTM that explicitly integrates maneuver features—left turns, right turns, and straight-through movements—into an encoder–decoder forecasting framework. By encoding turning intent alongside kinematic states, the model reduces errors that typically occur during turning maneuvers, while maintaining accuracy for straight trajectories. Importantly, this work leverages high-resolution UAV-captured traffic data, which offers wide spatial coverage and minimizes occlusion compared to fixed-camera datasets. The richer, more continuous data stream enables more reliable labeling of turning behaviors and provides a realistic foundation for training and testing the proposed model.

Building on the above, we summarize our contributions as follows.

(i) It shows that explicit maneuver encoding—implemented via a 1-s cumulative heading change and one-hot turn indicators—stabilizes maneuver recognition and yields targeted accuracy gains where prediction is hardest: left and right turns. Improvements are concentrated on turning ADE/FDE while leaving straight-through performance essentially unchanged.

(ii) It presents a Turn-Aware LSTM that augments a standard encoder–decoder with turn features on the input side. The model is lightweight (≈2–3 ms per vehicle on an RTX 4090), matching vanilla-LSTM latency and thus suitable for real-time use.

(iii) It provides maneuver-resolved evaluations across 1–3 s horizons against CV, vanilla LSTM, and a Tiny Transformer. While the Transformer attains the lowest overall errors, the Turn-Aware LSTM consistently outperforms the vanilla LSTM—most notably for 3-s turning forecasts (~15–20% FDE reduction)—thereby isolating the benefit of maneuver encoding. The dataset, preprocessing pipeline, and threshold sensitivity (5°/10°/15°) are documented for reproducibility, and we discuss generalization beyond the study site as well as extensions to LiDAR/V2X fusion and turn-aware Transformer/GNN variants.

The remainder of this paper is structured as follows: Section 2 reviews existing trajectory prediction approaches, with particular attention to their limitations in handling turning maneuvers and intersection scenarios, thereby motivating the need for maneuver-aware forecasting models in this study. Section 3 details the research methodology, including UAV-based data collection at a signalized intersection, preprocessing techniques for stabilizing trajectories and encoding turning behaviors, and the design of the proposed Turn-Aware LSTM architecture. Section 4 reports the experimental results and comparative analysis, highlighting how maneuver-aware encoding improves prediction accuracy over baseline models and discussing challenges such as noise, stationarity, and long-horizon drift. Finally, Section 5 concludes the paper by summarizing key contributions, outlining implications for autonomous driving and traffic management, and suggesting directions for broader validation and multimodal extensions.

2. Literature Review

Vehicle behaviour detection is essential for traffic monitoring research, with most existing methods relying on vehicle trajectory analysis. Traditional machine learning techniques, such as Fuzzy C-Means (FCM) and Support Vector Machines (SVM), have been employed to classify vehicle trajectories. Saini et al. demonstrated the effectiveness of FCM and SVM in trajectory classification, although these methods exhibited limitations in feature robustness[6]. Similarly, Yao et al. [7] proposed a trajectory clustering framework that encodes trajectory depth as a fixed-length feature sequence, while Choong et al. [8] utilized the Longest Common Subsequence (LCSS) algorithm to measure trajectory similarity before clustering. Despite their contributions, these traditional methods struggle to scale effectively for large datasets and dynamic traffic conditions, limiting their applicability in real-world traffic analysis.

Deep learning models, particularly Long Short-Term Memory (LSTM) networks, have gained significant traction in vehicle behaviour detection due to their capability to handle sequential data and address vanishing gradient issues [9]. LSTMs have been successfully applied in behaviour recognition, including robot behaviour classification and abnormal behaviour detection in video sequences [10]. In the context of vehicle trajectory prediction, Morton et al. [11] utilized LSTMs to predict vehicle acceleration on highways, demonstrating superiority over traditional models. Further improvements were introduced by Ding et al. [12], who combined LSTM-based models with Convolutional Neural Networks (CNNs) to detect unsafe driving behaviours. These studies highlight the effectiveness of LSTM-based approaches for trajectory prediction but also expose the limitations of current models in capturing complex spatial-temporal dependencies, particularly in mixed-traffic environments.

To address the challenge of modelling interactions between multiple vehicles, researchers have integrated attention mechanisms and graph-based models into LSTM architectures. The Spatiotemporal Attention Long Short-Term Memory (STA-LSTM) model, introduced by Lei Lin et al. [4], incorporates spatial-temporal attention mechanisms to enhance vehicle trajectory prediction by identifying how historical trajectories and surrounding vehicles influence future movement. Similarly, Yang and Pei [13] developed the Long-Short Term Spatio-Temporal Aggregation (LSSTA) network, which combines transformer networks with Temporal Convolution Networks (TCN) to improve long-term dependency modelling in vehicle behaviour prediction. Despite these advances, many models continue to depend on external data sources, such as GPS and road geometry, making them less adaptable to real-world scenarios.

Our study introduces a Turn-Aware LSTM model that exclusively relies on vehicle trajectory data to identify lane positions and predict turning behaviours, eliminating the need for GPS or road geometry inputs. This approach directly addresses the limitations in existing research by emphasizing lane-specific interactions and turning behaviours to improve trajectory prediction accuracy. The customized YOLOv8 model was trained on high-resolution drone footage collected at intersections, allowing for enhanced vehicle detection and tracking [14]. By analysing cumulative directional changes and displacement patterns, our model classifies turning behaviours based on angular variations, providing a robust framework for real-time vehicle behaviour prediction.

The reliance on fixed roadside cameras, such as those used in the NGSIM dataset [15], has posed challenges for vehicle trajectory prediction due to occlusions, limited spatial resolution, and constrained fields of view. These limitations can lead to incomplete or inaccurate vehicle detection, particularly in high-density traffic and intersection scenarios.

The existing literature establishes a strong foundation for vehicle trajectory prediction using both traditional and deep learning methods. While LSTMs and transformer-based architectures have demonstrated promising results, most existing models continue to depend on external infrastructure-based data, limiting their applicability in mixed-traffic environments [16][17]. More importantly, they often treat vehicle motion as a uniform process, without distinguishing between different maneuver types. In practice, however, intersection studies consistently show that left and right turns are far more error-prone than straight-through movements, both in terms of prediction drift and safety risk [18][19][20]. Forecasting models that ignore these maneuver-specific differences may achieve reasonable accuracy for straight trajectories but often show reduced reliability during turning—arguably the very moments when accurate prediction is most critical for avoiding conflicts.

In addition, many methods classify turns frame by frame, making them vulnerable to jitter, occlusion, or noise in tracking [21][22]. This often leads to unstable maneuver recognition, particularly during stationary periods or at the onset of a turn, thereby reducing the reliability of downstream prediction. While prior work has laid an important foundation for vehicle trajectory prediction, turning maneuvers remain especially challenging due to their variability and sensitivity to noise.

To complement these efforts, this study introduces a Turn-Aware LSTM that incorporates cumulative turning-angle encoding—a modest but practical refinement aimed at improving robustness for left and right turns while preserving accuracy for straight movements.

By explicitly representing left, right, and straight maneuvers through one-hot vectors, the model achieves more stable recognition and delivers improved trajectory prediction precisely where conventional models tend to degrade: during turning maneuvers and stationary phases at intersections.

3. Methodology

Traffic behavior detection plays a critical role in transportation research, particularly in the field of vehicle trajectory prediction. Traditional methods rely on trajectory analysis, using classical machine learning techniques such as Fuzzy C-Means (FCM) and Support Vector Machines (SVM) for classification. However, these methods struggle with feature robustness and scalability, particularly in dynamic traffic conditions. Recent advances in deep learning, particularly Long Short-Term Memory (LSTM) networks, have significantly improved the accuracy of trajectory prediction by modeling sequential dependencies in vehicle movement data.

While existing LSTM-based models have demonstrated high accuracy in specific scenarios, they often fail to incorporate essential contextual features such as lane-specific functions and turning behaviors, which are critical for predicting vehicle movements in complex intersection environments.

To address these limitations, this study introduces a Turn-Aware LSTM model that explicitly encodes maneuver types (straight, left turn, right turn) alongside kinematic trajectory features. This design enables the model to learn differentiated motion dynamics across maneuvers, thereby improving prediction accuracy in exactly the situations where conventional models degrade.

The proposed methodology consists of five main components: data collection and preprocessing, vehicle detection and tracking, trajectory forecasting, turning behavior recognition and Turn-Aware LSTM. To ensure consistent scaling and facilitate model convergence, all numerical features were normalized before being fed into the Turn-Aware LSTM.

3.1 Data collection and preprocessing

As shown in Fig. 1, the study site is a four-arm signalized intersection in Châteauguay, Montreal. The layout includes two westbound lanes (W1–W2), three eastbound lanes (E1–E3), three southbound lanes (S1–S3), and three northbound lanes (N1–N3). The intersection design provides dedicated turning lanes that are clearly separated from through lanes under signal control. This configuration makes the site suitable for isolating turning maneuvers in the collected vehicle trajectories. Data were recorded on December 7th and 8th, 2020, during both peak and off-peak periods to capture diverse traffic conditions. The choice of this location reflects the study’s focus on turning maneuvers, as the intersection’s lane geometry—with dedicated turning lanes under signal control—provides a clear setting for examining how turning behavior can be explicitly modeled in trajectory forecasting.

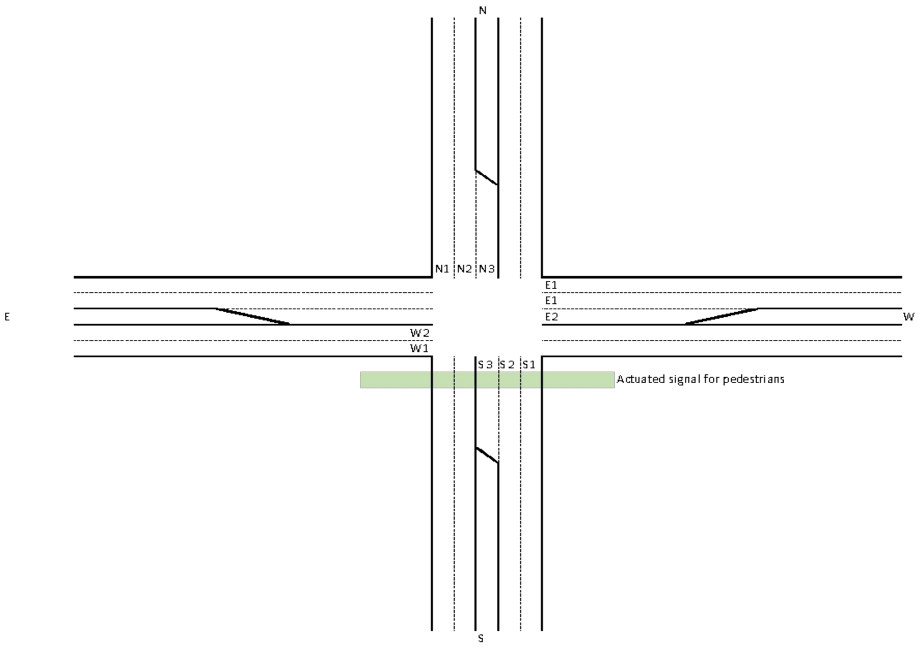

Fig. 1 - Layout of the four-arm signalized intersection in Châteauguay, Montreal, QC

Fig. 2 presents the temporal distribution of traffic during the data collection period, reflecting the peak-hour congestion observed at the study site. The dataset comprises vehicles classified into passenger vehicles, trucks, and buses, each labelled with distinct identifiers for subsequent tracking. Video recordings were captured in 4K resolution at 30 fps, ensuring high-fidelity motion analysis.

Fig. 2 - Aerial view of the study area in Châteauguay QC

Fig. 3 illustrates the UAV's position at an altitude of approximately 80 meters above the intersection, which served as the basis for pixel-to-distance calibration. The altitude was programmatically fixed and remained stable throughout data collection. The main source of variability came from minor lateral drift or tilt caused by wind or GPS fluctuations. At this elevation, each pixel corresponded to about 3.5 meters, providing a critical conversion parameter for vehicle trajectory computation. To mitigate distortions, a Fourier–Mellin transform (FMT) was applied for video stabilization, correcting translational, rotational, and small-scale deviations across frames. This ensured reliable trajectory extraction, although residual perspective errors may persist at the image periphery, a limitation acknowledged in Section 5.

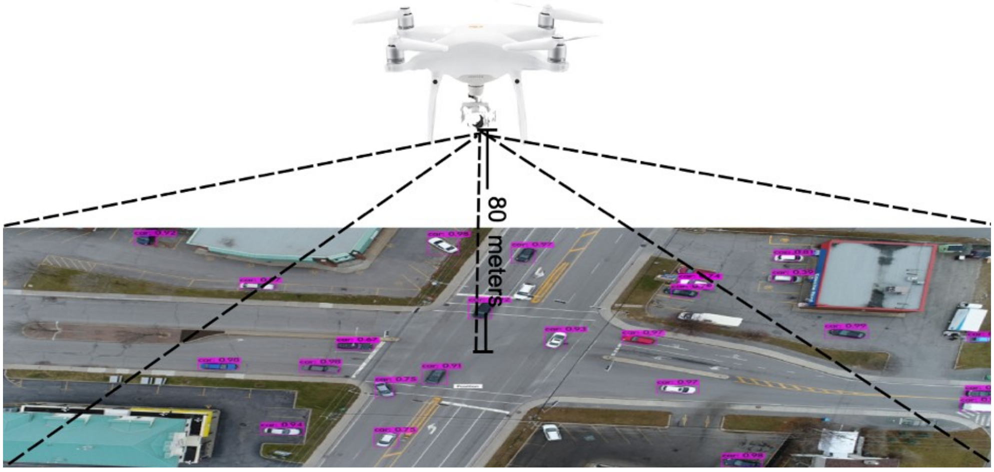

Fig. 3 - Drone positioned 80 meters above the intersection

To improve the consistency of trajectory extraction, a Fourier-Mellin transform was applied for video stabilization, mitigating translational and rotational deviations caused by UAV movement. This method effectively corrects distortions in video frames, ensuring continuous and reliable vehicle trajectories.

The Fourier-Mellin transform (FMT) leverages the Fourier rotation and similarity theorems to convert rotation and scaling into translations in log-polar space. Suppose two frames are related by translation and rotation, the transformation can be expressed as Equation (1)

$$f_2(x,y)=f_1(x\cos\theta+y\sin\theta-t_x,-x\sin\theta+y\cos\theta-t_y) \quad (1)$$

where:

• θ is the rotation angle,

• t_x and t_y are translations in the x and y directions, respectively.

This transformation allows for effective compensation of rotation and scaling changes between consecutive frames, ensuring that detected vehicle trajectories remain continuous and unaffected by UAV movements.

To formally define the Fourier-Mellin transform of a function, we use *Equation (2)*:

$$M_f(u,v)=\frac{1}{2}\int_0^{\infty}\int_0^{2\pi} f(r,\theta)r^{-ju}e^{-jv\theta}\,d\theta dr \quad (2)$$

where:

• u and r are the Mellin transform parameters,

• v and θ are the Fourier transform parameters.

This transformation remaps the Fourier-transformed frame into log-polar coordinates, allowing rotation and scaling to be expressed as simple translations. By applying this correction, rotational and scaling misalignments in UAV video frames are eliminated, resulting in stabilized sequences that accurately reflect real-world vehicle movements.

Fig. 4 illustrates the video frames before and after the Fourier-Mellin transform, highlighting the reduction in rotation and scaling effects. As seen in Fig. 5, color composite analysis visually confirms the improved alignment across successive frames.

Fig. 4 - Comparison of Original vs. Stabilized Frames (Fourier-Mellin Transform)

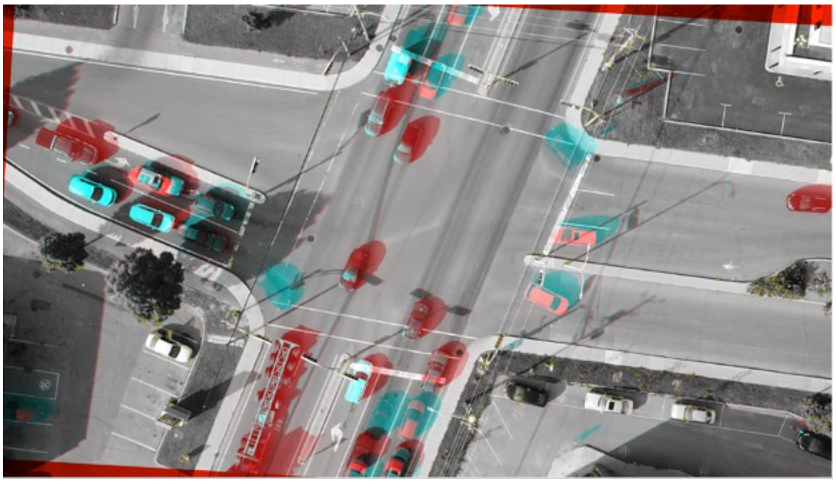

Fig. 5 - Frame Alignment via Color Composite Analysis

Additionally, background subtraction techniques were employed to filter out static objects, reducing noise and enhancing the accuracy of subsequent vehicle detection and tracking.

3.2 Vehicle detection and tracking

For object detection, YOLOv8 was employed due to its high precision in urban environments. The model was retrained on a custom dataset consisting of 18,000 labeled images to adapt to site-specific vehicle characteristics. Labeling was conducted using Open Labeling, assigning unique class identifiers (0 to 4) to different vehicle types (passenger cars, buses, and trucks). Given the frequent presence of stationary vehicles at intersections, dataset redundancy was minimized by selecting frames at 150-frame intervals, ensuring a balanced dataset. To further mitigate class imbalance, data augmentation techniques were applied, including rotation, cropping, scaling, and flipping, thereby improving the model's robustness in detecting underrepresented turning behaviors.

Vehicle trajectories were extracted using Deep SORT, which assigns unique tracking IDs across frames based on detections from YOLOv8. Each detected vehicle was represented by its bounding box parameters $\left(x_{center}, y_{center}\right)$, width, height, confidence score, and class label. Non-Maximum Suppression (NMS) was applied to remove redundant detections using the Intersection over Union (IoU), as Equation (3):

$$IoU\left(B_i, B_j\right) = \frac{\text{Area}\left(B_i \cap B_j\right)}{\text{Area}\left(B_i \cup B_j\right)} \tag{3}$$

where overlapping bounding boxes exceeding a threshold θ were suppressed to maintain detection accuracy.

3.3 Trajectory forecasting

Following vehicle trajectory extraction, a data cleaning pipeline was implemented to remove outliers and interpolate missing detections. The comprehensive end-to-end workflow, from raw data collection to the final LSTM-based forecasting, is summarized in Figure 6.

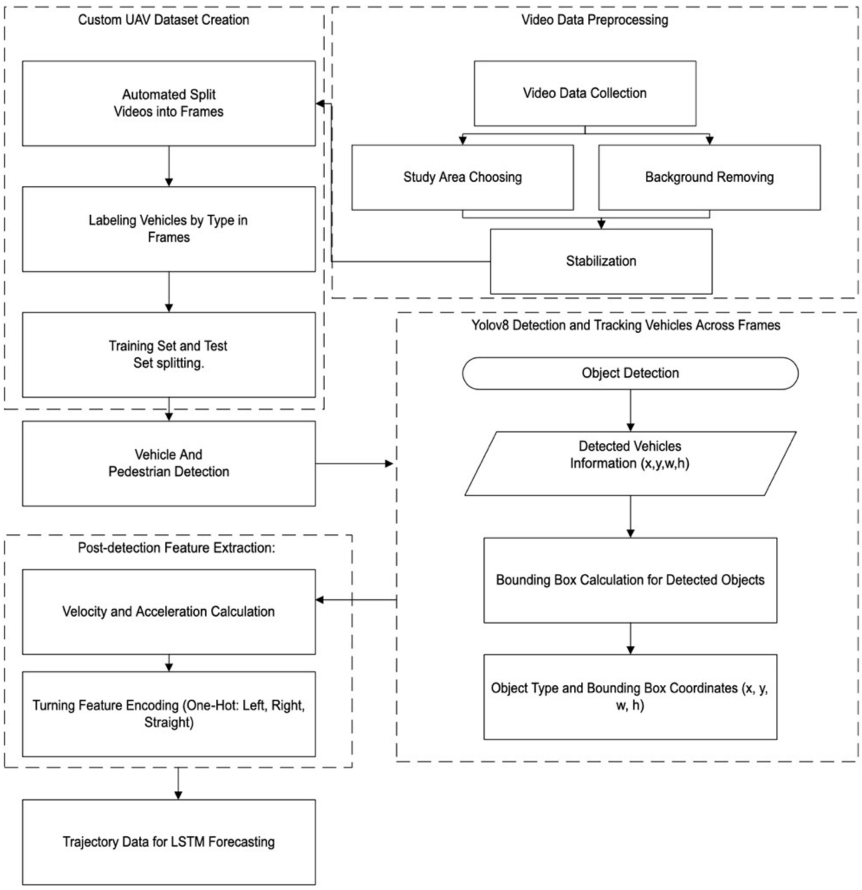

Fig. 6 - End-to-End Vehicle Trajectory Forecasting Model

After detection, vehicle velocities were computed using finite differences as Equation (4):

$$v_{x,t} = \frac{x_t - x_{t-1}}{\Delta t}, v_{y,t} = \frac{y_t - y_{t-1}}{\Delta t} \quad (4)$$

and acceleration components were estimated as Equation (5):

$$a_{x,t} = \frac{v_{x,t} - v_{x,t-1}}{\Delta t}, a_{y,t} = \frac{v_{y,t} - v_{y,t-1}}{\Delta t} \quad (5)$$

where Δt represents the time interval between frames. To ensure smooth and consistent velocity and acceleration estimates, a sliding window approach was applied to reduce fluctuations. Additionally, forward and backward interpolation was used to handle missing detections, ensuring seamless trajectory continuity.

To reduce noise, a Savitzky-Golay filter was applied separately to the x and y coordinates, preserving motion dynamics while smoothing trajectories over a sliding window. The Savitzky-Golay filter operates by fitting a polynomial of degree k over a window of $2m+1$ points for each coordinate, producing smoothed values x_i^{smooth} and y_i^{smooth} , as shown in Equation (6) and Equation (7).

$$x_i^{\text{smooth}} = \sum_{j=-m}^{m} c_j x_{i+j} \tag{6}$$

$$y_i^{\text{smooth}} = \sum_{j=-m}^{m} c_j y_{i+j} \tag{7}$$

3.4 Turning behavior recognition

After trajectory preprocessing, turning behaviors were extracted and encoded to capture lane-change and maneuvering actions, which are essential for accurate vehicle trajectory forecasting.

The cumulative turning angle over each 1-second window was used to classify turning behavior as left, right, or straight.

To capture turning behaviors, an instantaneous direction angle was calculated using Equation (8):

$$\phi_t = atan2\left(v_{y,t}, v_{x,t}\right) \tag{8}$$

Where $v_{x,t}$ and $v_{y,t}$ represent the velocity components. The angular change $\Delta\phi_t$ between consecutive frames was then computed as Equation (9):

$$\Delta\phi_t = \phi_t - \phi_{t-1} \tag{9}$$

A turning threshold of $\Theta = 10^\circ$ (0.1745 rad) was adopted to classify maneuvers into left, right, and straight categories. Heading-change thresholds are widely used in trajectory segmentation, where turns are identified once cumulative direction changes exceed a predefined angle [23][24][25]. Prior work shows that thresholds in the range of 5°–15° are commonly applied to distinguish between lane-keeping and turning behavior, although the exact value depends on data resolution and noise characteristics. To verify robustness, we conducted a sensitivity analysis across three thresholds of 5°, 10°, and 15°, and trajectory-level accuracy remained stable at ~83% across all settings.

Tab. 1 Sensitivity analysis results across three thresholds

Threshold (°)	Traj Overall	Left Acc	Straight Acc	Right Acc	Curve Recall	Straight FP	Onset Delay	Label Flips	Tracks Evaluated

	Acc						(s)	/100f	
5	0.835	0.807	0.874	0.833	0.203	0.44	0.0	21.08	260
10	0.835	0.807	0.874	0.833	0.2	0.439	0.0	20.96	260
15	0.835	0.807	0.874	0.833	0.197	0.439	0.0	20.86	260

This stability supports the adoption of 10° as a balanced threshold: small enough to detect true turning maneuvers, yet conservative against noise in frame-to-frame heading fluctuations. Based on this analysis, we operationalized the 10° rule at the frame level. At each time step t, the instantaneous direction angle ϕ_t was compared against the threshold to assign a preliminary label of left, right, or straight, as shown in Equation (10):

$$m_t = ¿ \qquad (10)$$

To mitigate noise at the frame level, the maneuver type of each trajectory was determined through aggregation rather than relying on instantaneous labels. Specifically, we applied a majority-vote strategy over the tail segment of the trajectory and required a minimum number of consecutive frames to confirm a turning event. This design ensures sensitivity to the onset of a turn while avoiding spurious fluctuations that may occur during short stops or tracking jitter.

Because LSTM models require numerical input, maneuver categories were encoded using a one-hot representation. This choice avoids introducing artificial ordinal relationships among the three classes (left, right, straight) and ensures that each maneuver is treated as an independent behavioral mode. The encoded features were then concatenated with the kinematic states and passed into the trajectory forecasting model, providing the LSTM with both motion history and maneuver context.

The final feature vector at each time step t was structured as Equation (11):

$$u_t = \left[x_t, y_t, v_{x,t}, v_{y,t}, a_{x,t}, a_{y,t}, m_{t,1}, m_{t,2}, m_{t,3}\right] \qquad (11)$$

where $m_{t,1}, m_{t,2}, \wedge m_{t,3}$ corresponded to the one-hot encoded turning classifications. The full input matrix for a vehicle's trajectory sequence was as Equation (12):

$$U_i = \left[u_{t_1}, u_{t_2}, \ldots, u_{t_n}\right] \qquad (12)$$

where n represents the number of time steps in the observation window.

Before feeding the trajectories into the prediction model, we performed consistency checks and normalization to ensure that the input data reflected physically plausible vehicle motion. This step was necessary because raw UAV-tracked trajectories can contain noise, missing detections, or unrealistic fluctuations. To address these issues, we applied the following procedures:

First, we identified and corrected implausible motion artifacts such as sudden position jumps, abrupt heading changes, and excessive acceleration spikes. Missing detections were handled by interpolation, either by propagating valid past or future values or by linear interpolation when both sides were available. After filling gaps, trajectories were resampled to the original frame rate of 30 fps to provide the temporal resolution required for sequential modeling.

Because frame-level turning cues are often noisy, turning-behavior encoding was performed on down-sampled one-second intervals to smooth jitter, and subsequently up-sampled back to 30 fps for synchronization. Finally, all numerical features (positions, velocities, accelerations) were normalized with respect to their maximum observed values to facilitate model convergence and comparability across trajectories.

With preprocessed and normalized features in place, we proceeded to the Turn-Aware LSTM architecture, which integrates these inputs into an encoder–decoder framework for sequence-to-sequence trajectory forecasting.

3.5 Turn-Aware LSTM

The proposed predictor utilizes an encoder–decoder LSTM architecture for sequence-to-sequence trajectory forecasting. The encoder comprises a stacked two-layer LSTM with a hidden size of 128 units to process the observed sequence and compress temporal dependencies into final hidden and cell states. Initialized with these states, the decoder autoregressively generates the future trajectory for the specified prediction horizon. At each decoding step, a linear projection maps the hidden state to 2D coordinates x and y. Training employs the Mean Squared Error loss and the Adam optimizer with a learning rate of 0.001. To enhance generalization, an early stopping mechanism is implemented if the validation loss remains stagnant for ten consecutive epochs.

4. Experiment

4.1 Experimental Setup

The dataset was divided into training (70%), validation (15%), and test (15%) sets, ensuring trajectories from the same vehicle remained within a single subset. Performance was evaluated at horizons of 1.0 s, 2.0 s, and 3.0 s (30, 60, and 90 frames). The Turn-Aware LSTM was compared against three baselines: (i) a physics-based Constant Velocity (CV) model; (ii) a conventional Vanilla LSTM; and (iii) a Tiny Transformer representing the state-of-the-art. This setup isolates the targeted accuracy gains of explicit maneuver encoding during turning maneuvers.

4.2 Feature Engineering and Visualization

To illustrate the impact of maneuver encoding, Figure 7 presents vehicle trajectories categorized into left turns, right turns, and straight movements. All trajectories were pre-processed with y-coordinate inversion to maintain consistency with the tracking system. These encoded behaviours serve as essential inputs, enabling the LSTM to capture complex maneuvering patterns and improve anticipation of vehicle movements compared to raw trajectory data.

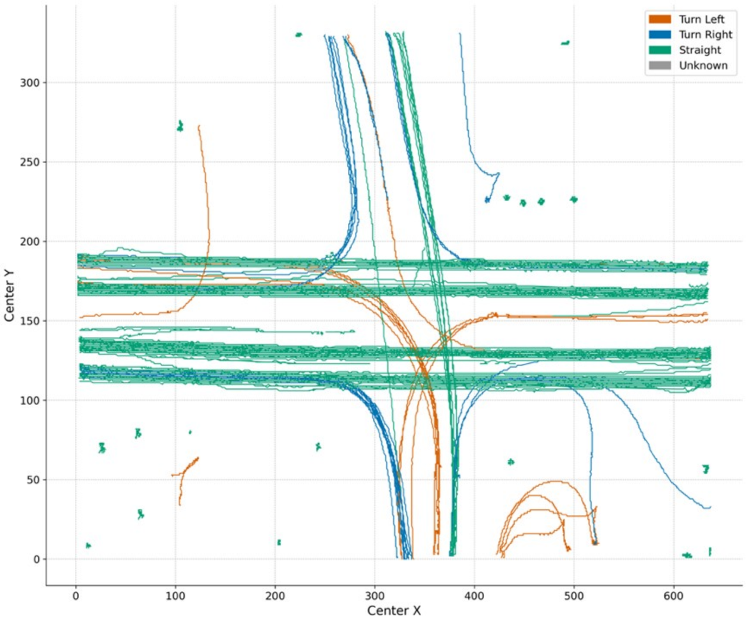

Fig. 7 Vehicle trajectories after turning feature encoding, categorized into left turns, right turns, and straight movements.

4.3 Comparative Analysis across Models and Maneuvers

As illustrated in Figure 8 and Figure 9 prediction errors increase with longer horizons due to accumulated uncertainty. The Tiny Transformer achieves the lowest overall Average Displacement Error (ADE) and Final Displacement Error (FDE), while the Turn-Aware LSTM consistently outperforms the Vanilla LSTM, particularly at the 3s horizon (0.30 m vs. 0.35 m FDE). The CV baseline exhibits unstable and significantly higher errors across all metrics, confirming the necessity of sequence modelling for intersection dynamics.

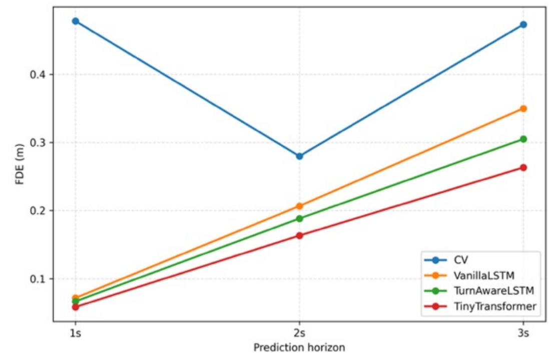

Fig. 8 Average FDE across all maneuvers over different prediction horizons

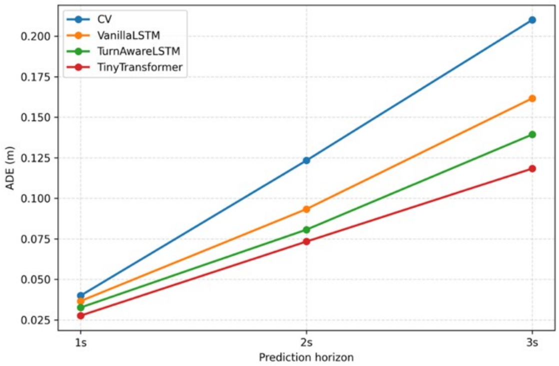

Fig. 9 Average ADE across all maneuvers over different prediction horizons

The targeted benefit of turn encoding is highlighted in Figure 10 and Figure 11. For right turns, the Turn-Aware LSTM cuts the extreme drift produced by the CV model almost in half, reaching ~0.42 m at 3 s. For left turns, the model reduces error from ~0.35 m (Vanilla LSTM) to ~0.28 m at 3 s, validating the use of maneuver-aware features in curbing curvature misestimation. Performance on straight motion remains comparable across all learning-based models, confirming that turn features do not interfere with trivial forward motion. Inference times on an RTX 4090 were ~2.5 ms per trajectory, confirming the model's suitability for real-time deployment.

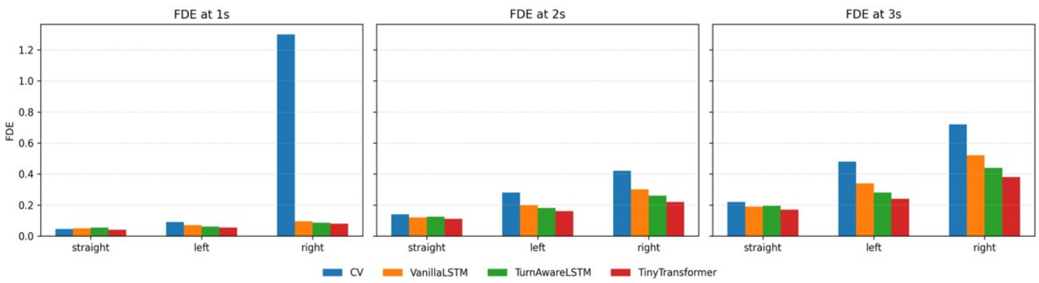

Fig. 10 FDE across all maneuvers over different prediction horizons

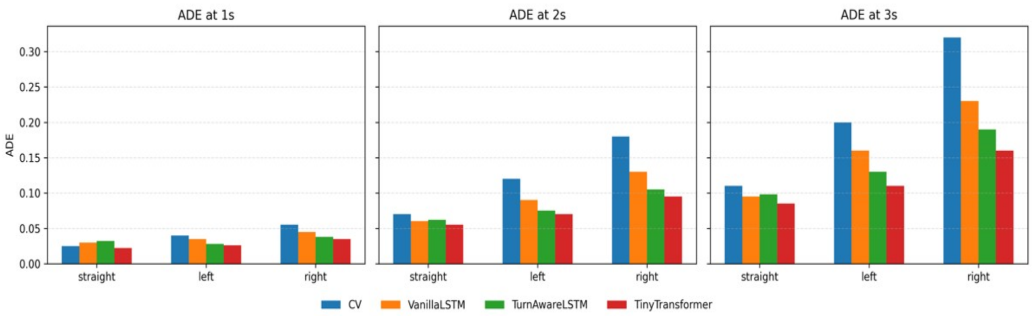

Fig. 11 ADE across all maneuvers over different prediction horizons

In addition to accuracy, we also evaluated computational efficiency. On an RTX 4090 GPU, inference times were ~2.5 ms per trajectory for the Turn-Aware LSTM, ~2.3 ms for Vanilla LSTM, and ~4.8 ms for Tiny Transformer, while the CV baseline was nearly instantaneous (<1 ms). This confirms that the proposed turn-aware encoding introduces only minimal overhead compared to a standard LSTM, while remaining well within the latency requirements for real-time deployment.

4.4 Experiment Discussion

Turning maneuvers remain the most challenging to predict due to detection jitter, which can cause spurious angular fluctuations during stationary periods [26]. Our experimental analysis (formerly illustrated via frame-level labelling) revealed that vehicles waiting at red lights could be misclassified as executing rapid turns in consecutive frames due to minor tracking noise. To address this, we adopted a cumulative turning angle strategy that aggregates directional changes over a 1-second window. This approach successfully smooths frame-level jitter and ensures that the reported performance gains, such as the 20% FDE reduction for left turns, reflect genuine improvements in trajectory prediction rather than artifacts of noisy labelling.

Figure 12 provides a representative snapshot of the real-time pipeline, confirming the model's robustness. For instance, vehicles executing left and right turns (e.g., IDs 11, 50, and 1) follow the ground truth paths closely without the curvature drift typical of vanilla LSTMs. Crucially, the cumulative-angle method correctly identifies stationary vehicles (e.g., IDs 24 and 26) as stopped, maintaining stable forecasts even under detection fluctuations. With a minimal computational overhead of ~2.5 ms per trajectory, the Turn-Aware LSTM demonstrates a practical balance between accuracy and real-time efficiency.

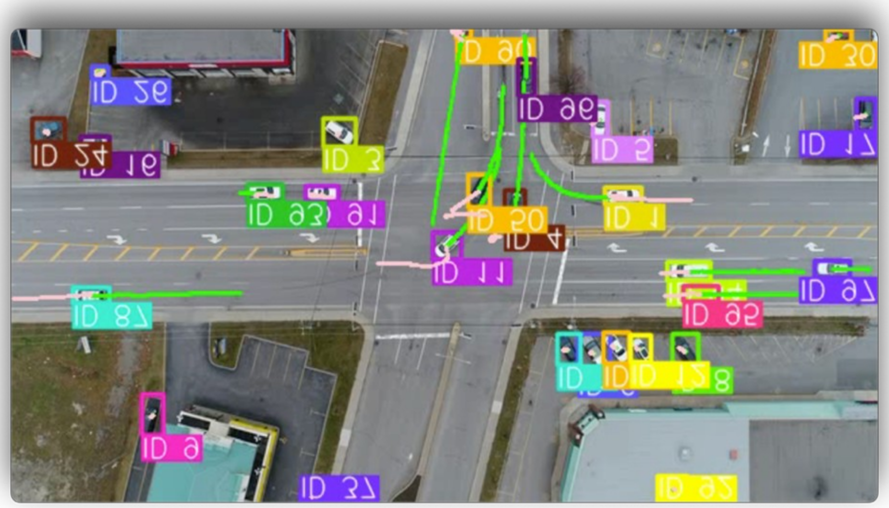

Fig. 12 Forecasted and ground truth trajectories visualization

5. Conclusion

This study demonstrates that explicitly encoding turning behaviours via cumulative heading changes improves vehicle trajectory forecasting, reducing FDE by 15–20% for turning maneuvers compared to a vanilla LSTM. The model's lightweight design (~2.5 ms inference) makes it suitable for real-time autonomous driving and proactive traffic safety management without reliance on GPS or detailed maps. While the Tiny Transformer achieves lower long-horizon errors, the Turn-Aware LSTM provides targeted improvements where they matter most—at turns. Future research will focus on integrating maneuver encoding into graph-based interaction models and validating the framework on larger datasets like Waymo to improve robustness in high-density, multi-agent traffic environments.

5.1. Data Availability Statement

The processed trajectory datasets, maneuver annotations, and model code are available at the following repository: https://github.com/Jynxzzz/Turn-Aware-LSTM_SUPP. Due to privacy and data-sharing restrictions, raw video data cannot be publicly released.

6. Acknowledgments

This work was funded and supported by Ericsson - Global Artificial Intelligence Accelerator AI Hub Canada in Montréal through the Mitacs Accelerate Program. The authors would like to express their gratitude for the valuable comments and collaboration provided by GAIA Montréal, as well as the support from all collaborators involved in this study.

7. References

1. Tilg, G., Krause, S. and Bogenberger, K. (2021). 3. automatisierte Fahrzeuge, vernetzte Fahrzeuge, Managed Lanes, Kapazität. 2021.
2. Yan, X., He, J., Wu, G., Sun, S., Wang, C., Fang, Z. and Changjian, Z. (2024). 2. Driving risk identification of urban arterial and collector roads based on multi-scale data. *Accident Analysis & Prevention*.
3. Liao, H., Li, Z., Wang, C., Shen, H., Liao, D., Wang, B., … Xu, C.-Z. (2024). MFTraj: Map-Free, Behavior-Driven Trajectory Prediction for Autonomous Driving.
4. Lin, L., Li, W., Bi, H. and Qin, L. (2021). Vehicle Trajectory Prediction Using LSTMs with Spatial-Temporal Attention Mechanisms. *IEEE Intelligent Transportation Systems Magazine*, pp. 0–0.
5. Wang, J., Liu, K. and Li, H. (2024). LSTM-based graph attention network for vehicle trajectory prediction. *Computer Networks*.
6. Saini, R., Kumar, P., Roy, P. P. and Dogra, D. P. (2017). An efficient approach for trajectory classification using FCM and SVM. July 2017.
7. Yao, D., Zhang, C., Zhu, Z., Huang, J. and Bi, J. (2017). Trajectory clustering via deep representation learning. May 2017.
8. Choong, M., Angeline, L., Chin, R., Yeo, K. B. and Teo, K. (2017). Modeling of vehicle trajectory clustering based on LCSS for traffic pattern extraction. October 1, 2017.
9. Hochreiter, S. and Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9, n. 8, pp. 1735–1780.
10. How, D. N. T., Loo, C. K. and Sahari, K. S. M. (2016). Behavior recognition for humanoid robots using long short-term memory. *International Journal of Advanced Robotic Systems*, 13, n. 6, pp. 1729881416663369.

11. Morton, J., Wheeler, T. A. and Kochenderfer, M. J. (2017). Analysis of Recurrent Neural Networks for Probabilistic Modeling of Driver Behavior. *IEEE Transactions on Intelligent Transportation Systems*, 18, n. 5, pp. 1289–1298.
12. Ding, L., Fang, W., Luo, H., Love, P. E. D., Zhong, B. and Ouyang, X. (2018). A deep hybrid learning model to detect unsafe behavior: Integrating convolution neural networks and long short-term memory. *Automation in Construction*, 86, pp. 118–124.
13. Yang, C. and Pei, Z. (2023). Long-Short Term Spatio-Temporal Aggregation for Trajectory Prediction. *IEEE Transactions on Intelligent Transportation Systems*, 24, pp. 4114–4126.
14. Varghese, R. and M, S. (2024). YOLOv8: A Novel Object Detection Algorithm with Enhanced Performance and Robustness. 2024.
15. Coifman, B. and Li, L. (2017). A critical evaluation of the Next Generation Simulation (NGSIM) vehicle trajectory dataset. *Transportation Research Part B: Methodological*, 105, pp. 362–377.
16. Geng, M., Li, J., Xia, Y. and Chen, X. (Michael). (2023). A physics-informed Transformer model for vehicle trajectory prediction on highways. *Transportation Research Part C: Emerging Technologies*, 154, pp. 104272.
17. Ying, J. and Feng, Y. (2024). Infrastructure-Assisted cooperative driving and intersection management in mixed traffic conditions. *Transportation Research Part C: Emerging Technologies*, 158, pp. 104443.
18. Ghanim, M. and Shaaban, K. (2018). Estimating Turning Movements at Signalized Intersections Using Artificial Neural Networks. *IEEE Transactions on Intelligent Transportation Systems*, PP, pp. 1–9.
19. Zhang, S., Abdel-Aty, M., Cai, Q., Li, P. and Ugan, J. (2020). Prediction of pedestrian-vehicle conflicts at signalized intersections based on long short-term memory neural network. *Accident Analysis & Prevention*, 148, pp. 105799.
20. Schulz, J., Hubmann, C., Löchner, J. and Burschka, D. (2018). Interaction-Aware Probabilistic Behavior Prediction in Urban Environments. arXiv.
21. Gulzar, M., Muhammad, Y. and Muhammad, N. (2021). A Survey on Motion Prediction of Pedestrians and Vehicles for Autonomous Driving. *IEEE Access*, PP, pp. 1–1.
22. Hartjen, L., Philipp, R., Schuldt, F., Howar, F. and Friedrich, B. (2019). Classification of Driving Maneuvers in Urban Traffic for Parametrization of Test Scenarios. November 22, 2019.
23. Aboah, A., Adu-Gyamfi, Y., Gursoy, S. V., Merickel, J., Rizzo, M. and Sharma, A. (2023). Driver Maneuver Detection and Analysis Using Time Series Segmentation and Classification. *Journal of transportation engineering. Part A, Systems*, 149, n. 3, pp. 7312.
24. Phuyal, B. P. (2002). Turn detection algorithm for vehicle positioning.
25. Wan, Z., Li, L., Yu, H. and Yang, M. (2022). A Long Short-Term Memory-Based Approach for Detecting Turns and Generating Road Intersections from Vehicle Trajectories. *Sensors*, 22, n. 18, pp. 6997.
26. Zhao, X., Wang, G., He, Z. and Jiang, H. (2022). A survey of moving object detection methods: A practical perspective. *Neurocomputing*, 503, pp. 28–48.