Detection of Cyberattacks at Connected Signalized Intersections – From the Physical Perspective of Cyber-Physical Systems

Journal:	*IEEE Transactions on Intelligent Vehicles*
Manuscript ID	T-IV-23-09-2962
Manuscript Type:	Full Length Article
Date Submitted by the Author:	25-Sep-2023
Complete List of Authors:	Gu, Yingfan; University of Cincinnati, Department of Civil & Architectural Engineering & Construction Li, Zhixia; University of Cincinnati, Department of Civil and Architectural Engineering and Construction Management Zhang, Yunpeng; University of Houston, Department of Information Science Technology Tiwari, Shivam; University of Houston, Department of Information Science Technology Wei, Heng; University of Cincinnati, Department of Civil and Architectural Engineering and Construction Management Ma, Muting; The University of Alabama, Culverhouse College of Business Baidya, Sabur; University of Louisville, Department of Computer Science and Engineering
Keywords:	Machine Learning, System integration, safety and security;System design, modeling and deployment, Driver behaviour
Abstract:	Detecting cyberattacks is crucial for securing the connected transportation system. Previous research has predominantly focused on perspective of sensor intrusions and network anomalies. This type of detection from the cyber perspective of the cyber-physical systems may have the risk of the detection algorithm also being a target of the attacks. In contrast, standalone sensor-data-based algorithms that avoid communication to detect cyberattacks based on patterns of vehicle trajectories from the physical perspective of a cyber-physical system is promising to become a high-efficiency alternative to avoid the algorithm being attacked. This study develops a vehicle trajectory-oriented machine-learning algorithm to detect cyberattacks at a connected signalized intersection. Vehicle trajectories and driving behaviors are captured using a driving simulator under normal and cyberattack scenarios. Cyberattack detection models were constructed based on the collected datasets using Hidden Markov Models (HMMs), namely HMM-4-C. Whether a vehicle is being attacked or not is considered as a hidden state, and the trajectory data serve as the observation sequence. Expectation Maximization (EM) algorithm is applied to estimate the likelihood parameters of HMM-4-C. To verify the performance, an existing popular algorithm Bi-directional Long Short-Term Memory (Bi-LSTM) is built to facilitate a comparative analysis. The results indicate that 94.42% of cyberattack scenarios were successfully detected using HMM-4-C. It shows the proposed HMM-4-C outperforms the Bi-LSTM with an F1 score of 0.86 versus 0.71, respectively. In conclusion, the proposed method effectively detects future cyberattacks, which can be used to develop proactive cyberattack detection strategies in conjunction with cyber- perspective detection algorithms.

Detection of Cyberattacks at Connected Signalized Intersections – From the Physical Perspective of Cyber-Physical Systems

Yingfan Gu, Zhixia Li, Yunpeng Zhang, Shivam Tiwari, Heng Wei, Muting Ma and Sabur Baidya

[1]***Abstract*—Detecting cyberattacks is crucial for securing the connected transportation system. Previous research has predominantly focused on perspective of sensor intrusions and network anomalies. This type of detection from the cyber perspective of the cyber-physical systems may have the risk of the detection algorithm also being a target of the attacks. In contrast, standalone sensor-data-based algorithms that avoid communication to detect cyberattacks based on patterns of vehicle trajectories from the physical perspective of a cyber-physical system is promising to become a high-efficiency alternative to avoid the algorithm being attacked. This study develops a vehicle trajectory-oriented machine-learning algorithm to detect cyberattacks at a connected signalized intersection. Vehicle trajectories and driving behaviors are captured using a driving simulator under normal and cyberattack scenarios. Cyberattack detection models were constructed based on the collected datasets using Hidden Markov Models (HMMs), namely HMM-4-C. Whether a vehicle is being attacked or not is considered as a hidden state, and the trajectory data serve as the observation sequence. Expectation Maximization (EM) algorithm is applied to estimate the likelihood parameters of HMM-4-C. To verify the performance, an existing popular algorithm Bi-directional Long Short-Term Memory (Bi-LSTM) is built to facilitate a comparative analysis. The results indicate that 94.42% of cyberattack scenarios were successfully detected using HMM-4-C. It shows the proposed HMM-4-C outperforms the Bi-LSTM with an F1 score of 0.86 versus 0.71, respectively. In conclusion, the proposed method effectively detects future cyberattacks, which can be used to develop proactive cyberattack detection strategies in conjunction with cyber- perspective detection algorithms.**

***Index Terms*—Connected vehicle, cyberattack, machine learning, HMMs, HMM-4-C, LSTM, vehicle trajectories, driving behaviors.**

A. INTRODUCTION

The emergence of connected vehicles and connected and autonomous vehicle technologies is expected to pose cybersecurity issues to the urban transportation system. [1]–[4]. Connected vehicles are equipped with an On-Board Unit (OBU) that broadcasts Basic Safety Messages (BSMs), which include information such as the vehicle's location, speed, acceleration rate, etc. The roadside Unit (RSU) receives the BSMs and relays the information to other vehicles and infrastructure [5]–[8]. As vehicles become more and more connected to the wireless networks, this technological advancement expands the attack surface for cybercriminals to execute sophisticated cyberattacks [9].

The frequency of cyberattack events is on the rise [10]–[12]. For example, Two traffic engineers hacked into the signal system, and reduced traffic flows by programming the signals because of the labor protest [13]. New York City's wireless vehicles were attacked by a cybersecurity expert using a very cheap wireless device [14]. In Texas, hackers broke into a traffic sign and changed the messages on the digital signs of a road closure message with "Zombies Ahead" [15]. It is catastrophic if cyberattacks occur within the transportation system, with hackers potentially causing disastrous events [16], [17]. Furthermore, the Advanced Traffic Management System (ATMS) will be a critical component of the USA transportation system shortly. With the increasing use of advanced Traffic Management Systems, the United States is becoming an increasingly popular target for cyberattacks [18], [19].

When a cyberattack occurs, current technologies still find it difficult to distinguish the specifics of the

This work is partly supported by the USDOT Tier 1 UTC Transportation Cybersecurity Center for Advanced Research (CYBER-CARE). *(Corresponding author: Zhixia Li).*

Yingfan Gu is with the Civil & Architectural Engineering & Construction Department, University of Cincinnati, Cincinnati, Ohio 45221, USA (guyf@mail.uc.edu).

Zhixia Li is with the Civil & Architectural Engineering & Construction Department, University of Cincinnati, Cincinnati, Ohio 45221, USA (lizx@ucmail.uc.edu).

Yunpeng Zhang is with the College of Technology, Information and Logistics Technology Department, University of Houston-Main Campus, Houston, Texas 77204, USA (yzhan226@central.uh.edu).

Shivam Tiwari is with the College of Technology, Information and Logistics Technology Department, University of Houston-Main Campus, Houston, Texas 77204, USA (stiwari4@CougarNet.UH.EDU).

Heng Wei is with the Civil & Architectural Engineering & Construction Department, University of Cincinnati, Cincinnati, Ohio 45221, USA (heng.wei@uc.edu).

Muting Ma s with the Culverhouse College of Business, The University of Alabama, Tuscaloosa, Alabama, 35487, USA (mma10@ua.edu).

Sabur Baidya is with the Department of Computer Science and Engineering, The University of Louisville, Louisville, Kentucky, 40292, USA (shbaid01@louisville.edu).

scenario. Existing methodologies has predominantly focused on cyberattack detection from the perspective of sensor intrusions and network anomalies [20]–[22]. This type of detection of cyberattacks from a cyber perspective of the cyber-physical system, such as using algorithmic recognition of network attacks, carries the risk of also becoming a target when attacks occur. Compared to the cyber perspectives, using a standalone algorithm to analyze vehicle trajectories and driving behaviors patterns based on sensor data from the physical perspective of a cyber-physical system is promising to be useful for detecting irregularities that could indicate a cyberattack. Previous research has suggested that vehicle trajectories could be useful for studying cybersecurity [23]. If we could establish a standalone detection algorithm that is based on vehicle trajectories under cyberattacks from sensor data, early-stage warning for the cyberattack scenarios at the very beginning when the cyberattack happens would become feasible. This method that avoids communication is promising to become a high-efficiency alternative cyberattack detection method to avoid the detection algorithm being attacked. However, there has been a lack of such trajectory-data-based detection algorithms, as there lacks such ground truth trajectory data collected under real cyberattack at transportation facilities to train and validate the detection models.

In addition, to accurately detect cyberattack, we need to model vehicle trajectories in an efficient manner. Existing method to detect trajectory include: Convolutional Neural Network (CNN), Gaussian Mixture Model (GMM) or Recurrent Neural Network (RNN) [24]–[26]. Neural network-based methods have been proven effective in detecting anomalies in vehicle trajectories. They could potentially be used to address cybersecurity issues related to these trajectories. However, their accuracy may not always be optima. Neural network-based methods require a large amount of historical data for training. This might not be suitable in the context of cyberattacks, as there isn't an ample amount of existing data available for model training. Long Short-Term Memory (LSTM) is a widely used RNN architecture in trajectory estimation. Research has proved its effectiveness and accuracy in the area of pedestrians' trajectory prediction [27] and uncrewed aerial vehicles detection [28], [29]. It can potentially remember patterns over long sequences, but have a disadvantage when it comes to abrupt changes, like at the moment of an attack. During cyberattacks, the upcoming trajectory and behavior might be stochastic. Therefore, there is need of a suitable method to overcome the challenges of inadequate data and abrupt trajectory turning to accurately detect cyberattack events. As driver behavior is difficult to be predicted directly as the behavior is stochastic and depends only on the present state, making it suitable for description as a Markov process. Hidden Markov models (HMMs) can therefore be a potential tool because the upcoming trajectory and behavior are stochastic and depend on the last state, which can be naturally described by a hidden Markov process [30]. This might make them more appropriate for estimating sudden changes in situations when an attack occurs [31], [32]. making them beneficial in cyberattack situations where data is limited. Vehicle trajectories can be characterized by their specific kinetic or geometric parameters under different scenarios [33].

In summary, HMMs may offer the following advantages in cyberattack trajectory pattern recognition: (1) HMMs are based on a probabilistic framework which provides naturally handles uncertainties in cyberattack trajectories. (2) Compared to neural networks, HMMs can be computationally more efficient and reliable. (3) HMMs might require less data to estimate their parameters, making them preferable in scenarios with limited data. Additionally, the literature review did not find HHMs being used to construct a cyberattack model. Therefore, this research aims to address two main issues. The first issue is the scarcity of available cyberattack scenarios, vehicle trajectories, and driver behavior for current researchers. To address this, we create a comprehensive cyberattack scenario within a connected vehicle environment. The second issue pertains to the absence of a method for detecting cyberattack scenarios. In response to this, we propose a state-of-the-art, field-based method named HMM-4-C to detect cyberattacks.

II. LITERATURE REVIEW

Cyberattacks increasingly threaten the stability of the transportation system and cause serious problems. Previous research primarily centered on vehicle platoon dynamics. Wang presented an extended car platoon model by introducing two additional weight parameters to characterize different security attacks [34]. Cui evaluated the effects of cyberattacks under cooperative adaptive cruise control (CACC) platoon scenarios using metrics such as speed variation, headway ratio, and injury probability [35]. The studies demonstrated that congestion attack will cause a big jam in the whole traffic system. Chen conducted the congestion attack on the U.S. Department of Transportation (USDOT)-sponsored design and implementation of a system called Intelligent Traffic Signal System (I-SIG) [36]. Also, different interventions of cyberattacks caused different types of impacts on travel time values of the networks, which could significantly increase travel times [37]. Previous researchers also found that cyberattack leads to serious traffic safety problems. Amoozadeh found that an attack can cause significant instability of the CACC vehicle stream [38].

In recent years, research has focused on cyberattacks on connected vehicles and connected autonomous vehicles [39]. Li tested attack factors and communicated positions and speeds in a longitudinal CAV using a collision index [40]. This research disclosed the dangerous attack ways in the CAV environment. Wang developed the CAV dynamics platoon under a cyberattack scenario using different acceleration, speed, and position [41]. This research explained different cyberattack effects on the CAV dynamics model. The results indicate that a cyberattack could potentially lead to serious disruptions in vehicle platoons. He presented a severity assessment method to evaluate the potential impact of cyberattacks on autonomous vehicles, which includes categorizing the various target assets into several levels [42]. Khattak conducted three types of cyberattacks, message falsification, dedicated denial of service, and spoofing attacks, and used the time to collision (TTC) to reveal that traffic stream and CAV string are unstable under cyberattacks [43]. Spoofing attacks on transportation facilities, such as traffic signal control systems and digital road signs, have also attracted the attention of scholars in recent years. Feng studied a cyberattack by manipulating the detector data in actuated and adaptive signal control. Different attack scenarios resulted in significant differences in the evaluation of system delay [44]. Perrine conducted an in-depth study to estimate the total delay of traffic when a cyberattack occurred on the traffic signal intersection, resulting in the signals being replaced with four-way stops. The results showed that the total delay was multiplied 4.3 times after 26 signals were attacked [45].

Consequently, detecting these cyberattacks has become an important research area in recent years. More recent attention has focused on the development of detection methods for cyberattack in connected vehicle environments. Huang simulated the attack by using falsified traffic trajectories. The authors introduced a fake vehicle that arrived and stopped at a specific location, presenting false data to the infrastructure. To detect falsified trajectory data, The authors proposed a method that involves training a neural network and conducting a hierarchical clustering algorithm. The results of the study indicated that their proposed method can effectively detect the presence of falsified trajectory data in V2I communication systems [6]. DeBruhl have designed attacks and abnormal behaviors to simulate cyberattack situations, and the successful detection of such attacks depends on the vehicle model switching when the misbehaviors occur [46]. The management scheme regarding attack-resistant trust (ART) has been proposed to detect attacks on wirelessly networked on the road to improve safety [47]. Biron presented a detection scheme for Denial of Service (DoS) attack in connected vehicle systems [48]. Mousavinejad presented a distributed attack detection algorithm that using predesign criteria of each vehicle state [49]. Ploeg conducted a simulation environment using cyberattack scenarios to test the vehicle's self-awareness in intersections. The results showed that the most relevant and potentially dangerous anomalies could be detected [50]. Xiao introduced a detection method that verifying the time-stamps of the data packets under the presence of intermittent communication attacks [51]. However, these previous researches have primarily focused on the perspective of sensor intrusions and network anomalies rather than the physical perspective. Understanding the behavior of human drivers during cyberattack events is crucial in comprehensively evaluating the impact of such attacks and developing effective countermeasures. Moreover, obtaining actual human driving data during cyberattacks is challenging. Most researchers used the hypothetical vehicle trajectory and assumed the vehicle movement under the event [51], [52]. When the attack occurred, the trajectories were strictly based on a pattern without humans getting involved. Previous research used assumed vehicle trajectory data and benchmark datasets for studying attack detection [53]–[55].

III. METHODOLOGY

A. The architecture of experimental scenarios

Signal countdown timer is a typical connected vehicle application that displays the remaining time of the red or green light on the driver's side [58]–[60]. When the vehicle approaches a connected intersection, the driver receives a red-light (RL) countdown information, allowing the driver to adjust the speed to cross the intersection when the traffic signal turns green. This application can substantially reduce the loss of speed by enabling the driver to adjust the speed of vehicles to go through the intersection, which can improve energy efficiency by maintaining a steady speed with fewer stops. Fig. 1 illustrates such situation. The vehicle will receive messages from infrastructure indicating the remaining red time and a screen on the vehicle dashboard will display the countdown information in seconds.

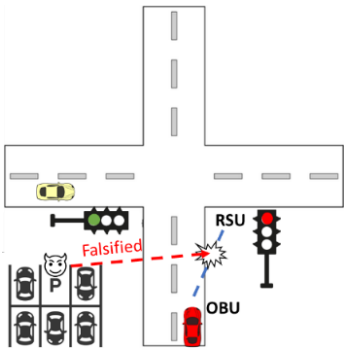
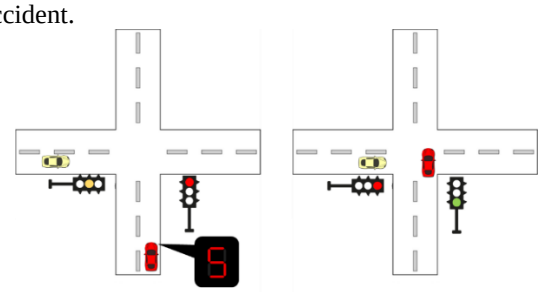


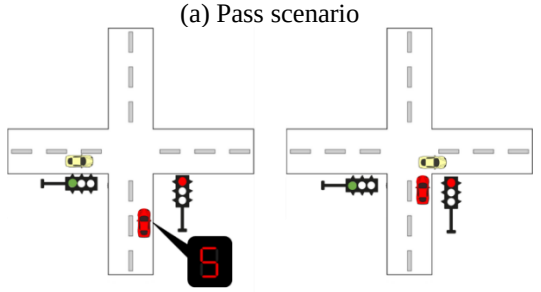
Fig. 1. Signal countdown timer attack model.

In this scenario, countdown messages will be sent from the traffic signal controller to the RSUs at the intersection. These RSUs will then broadcast the message to OBUs in the vehicles. From a cybercriminals' perspective, it's relatively difficult to attack the traffic signal controller to change Signal Phasing and Timing (SPaT) or the MAP, as the controller is physically located in a roadside box. However, launching an attack via the connection between RSUs and OBUs is much easier due to the potential for wireless attacks.

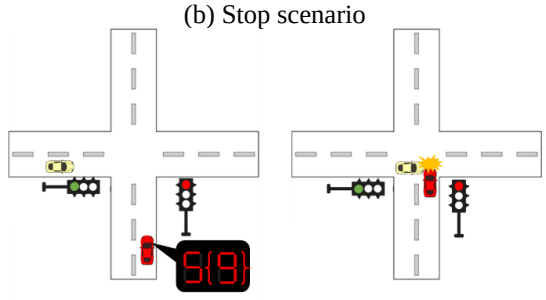
Spoofing wireless sensors is the most likely occurrence during an intersection cyberattack. In this case, cybercriminals may attack the system by changing or blocking messages between vehicles and infrastructure or inter-vehicle communication [61]. From a cybercriminals' perspective, red light running behavior is the most dangerous and could lead to more serious consequences. Therefore, a wireless attack on the red-light countdown application is the most likely scenario that cybercriminals would consider. If a cyberattack occurs, the driver will get a falsified message. The driver may still maintain the speed when the vehicle enters the intersection and result into a severe accident.



(a) Pass scenario



(b) Stop scenario



(c) Cyberattack scenario

Fig. 2. RL countdown scenarios at connected signalized intersection.

In this research, the signal countdown timer is chosen as the experiment the scenario. Generally, when a vehicle approaches an intersection where the red light is on, the driver will face two scenarios: the short-time RL and the long-time RL. A short-time RL means the driver could drive through the intersection without needing a full stop, or the light turns green before the drivers enter the intersection. Here we define this behavior as the Pass scenario as shown in Fig. 2a. A long-time RL means the driver must stop entirely before the stop line because it has a long red period. Here, we define this behavior as the Stop scenario as illustrated by Fig. 2b. In addition to the scenarios above, cybercriminals may attack by changing or blocking messages between vehicles and infrastructure. It is called the falsified data attack [62], [63]. If the cyberattack happens, the drivers will get a false message telling them there is a short RL period. The driver will still maintain the speed when the vehicle gets into the intersection. However, the actual RL period (9 seconds) differs from the falsified message on the dashboard (5 seconds). We define it as a Cyberattack scenario as illustrated by Fig. 2c.

B. Experiment procedure

Vehicle trajectory data under cyberattack is challenging to collect. In this research, we resort to simulating this specific scenario in the laboratory environment. The driving simulator creates an ideal environment for collecting vehicle trajectories. An experiment was designed to implement cyberattack scenarios under the connected vehicle environment. The miniSim driving simulator was applied in the research, presented by the National Advanced Driving Simulator (NADS).

A total of thirty-two participants with valid driving licenses were publicly recruited for the study, and their demographic information is summarized in Table 1. Each participant was required to drive through seven intersections three times, completing the experiment. Fig. 3 provides an overview of the route designed for the experiment, with the route indicated in red and the scenario zone in yellow. The connected vehicle application was implemented in all intersections. Figure 3 shows that the driver received the message indicating the remaining red-light time ("Red light remains in 5 seconds, 4 seconds… until 1 second") as they approached each intersection.

TABLE I

SUMMARY OF DEMOGRAPHICS FOR PARTICIPANTS (N=32)

Variable	Category	N	Percent (%)
Age	21~25 years old	23	71.9%
	26~30 years old	5	15.6%
	31~35 years old	4	12.5%
	36~40 years old	1	3.1%
Gender	Female	10	31.3%
	Male	22	69.7%

To minimize the impact of psychological expectations for the cyberattack scenario, the signal countdown timer in other intersections were kept functioning normally. The cyberattack scenario was only introduced at one intersection marked by the yellow area in Fig. 3. Each participant was required to drive through the entire route three times to complete the experiment. During the first time, this intersection was set to a short-time RL countdown (5 seconds), indicating the Pass scenario. During the second time, it was set to the Stop scenario, with a long-time RL countdown (9 seconds). For the third time, a cyberattack scenario we set up where cybercriminals falsified the RL countdown message. The long-time RL countdown message (9 seconds) was changed to a short-time countdown message (5 seconds. The red light was still on when the drivers reached the stop line at this intersection. Since the frequency of transmitting under the connected vehicle environment is 10Hz [64], we configured the data acquisition system to capture the vehicle's trajectory parameters at a sampling frequency of 10Hz. In the experiment, a speed limit of 35 mph was adopted.

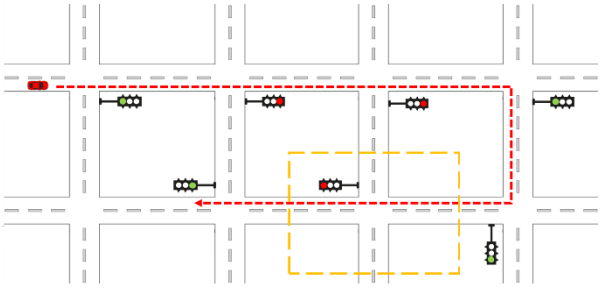
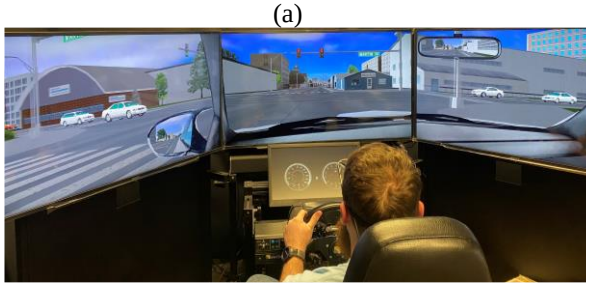


Fig. 3. Designed route of the experiment.

(a)



(b)

Fig. 4. Display RL countdown information and cyberattack scenairo at the intersection: (a) RL countdown; (b) Cyberattack scenairo.

C. *Hidden Markov models for vehicle trajectory (HMM-4-C) under the cyberattack*

This study treats vehicle movement as a time-series Markov process [65], [66], with different hidden states corresponding to the different scenarios in the cyberattack context. The time series of vehicle trajectories under each state is considered an observation [32], [67]. During the cyberattacks, the hidden state space is:

$$S = \{S_1, S_2, S_3\} \quad (1)$$

Where S_1 , S_2 and S_3 represents the Pass, Stop and Cyberattack scenario, respectively. The observations are the sequence of vehicle trajectory VT, recorded at a 10Hz time interval. M is the number of the observation.

$$VT = \{VT_1, VT_2, VT_3, \dots, VT_M\} \quad (2)$$

For a given observation sequence VT_i at time i, it can be represented by a matrix of size $N \times M$, N is number of recorded features. For vehicle trajectories, the characteristic features of the motion state are its kinetic state and direction of motion. In a cyberattack situation, changes in the kinetic state and motion direction indicate the impact of the cyberattack on vehicle trajectories. It will lead the attacked vehicle drive into the intersection or drive out of the road to causes catastrophic accident. In this research, we focus on changes in the axis direction connected to the intersection. The axis direction of velocity, acceleration and the distance to the stop line are regarded as the cyberattack features. M is the number of time intervals for which the observations are recorded. Therefore, VT_i can be represented by

$$VT_i = \begin{pmatrix} v_j \\ a_j \\ d_j \end{pmatrix}, \forall i = 1,2,3, \dots, M, \forall j = 1,2,3, \dots, N \quad (3)$$

Where v_j represents the velocity; a_j represents the acceleration; d_j represents the distance to the stop line.

Then, we denoted $\{Q\}$ as the hidden state segment during the time the vehicle is approaching the intersection. q_t means the one of hidden state in $\{S\}$ at time t when the vehicle is approaching. $\{O\}$ is denoted as the observations segment. The observation o_t is the observed vehicle trajectory in $\{VT\}$ for a simplified annotation. Figure 4 presents the hidden state and observation processes of HMM-4-C in this study.

$$Q = \{q_1, q_2, q_3, \dots, q_T\} \quad (4)$$

$$O = \{o_1, o_2, o_3, \dots, o_T\} \quad (5)$$

Equation (6) represents the probability of each state given the state attained in the previous time step [65], [66], which is

$$P(q_t|q_{t-1}, o_{t-1}, \cdots, q_1, o_1) = P(q_t|q_{t-1}) \quad (6)$$

In addition, the probability of each observation is independent of other observations and states and only depends on the current state.

$$P(o_t|q_t, o_{t-1}, \cdots, q_1, o_1) = P(o_t|q_t) \quad (7)$$

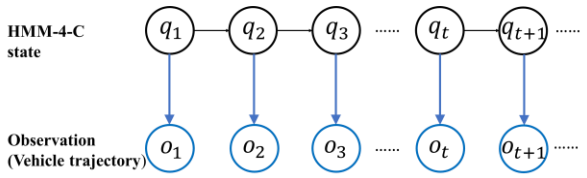


Fig. 5. The hidden state process and observation process of HMM-4-C.

The parameters of HMM-4-C can be represented by $\omega = (A, B, \pi)$ [6], [66], where A is the transition probability, B is the emission probability and π is the Initial probability of the state.

The transition probability A indicates hidden state transition likelihood matrix that can be represented by

$$A = [a_{ij}] \tag{8}$$

Each a_{ij} represent the probability of moving from hidden state i to state j.

$$a_{ij} = P(q_{t+1} = S_j | q_t = S_i),$$
$$\forall i, j = 1,2,3; \forall t = 1,2,3, \dots, T \tag{9}$$

The emission probability B represents the probability of each observation given a hidden state, and it can be represented by

$$B = [b_j(k)] \tag{10}$$

Each $b_j(k)$ means the probability of an observation k being generated from a given the current state j.

$$b_j(k) = P(o_t = VT_k | q_t = S_j),$$
$$\forall j = 1,2,3; \forall k = 1,2,3, \dots, M; \forall t = 1,2,3, \dots, T \tag{11}$$

The Initial probability of state π means the probability that the model will start in a state that can be represented by

$$\pi = [\pi_i] \tag{12}$$
$$\pi_i = P(q_1 = S_i), \forall i = 1,2,3 \tag{13}$$

Therefore, the probability of the state series $Q = \{q_1, q_2, q_3, \dots, q_T\}$ can be represented by

$$P(Q, \omega) = \pi_{q_1} \prod_{t=1}^{T-1} a_{q_t q_{t+1}} \tag{14}$$

The probability of the observation from the state series $Q = \{q_1, q_2, q_3, \dots, q_T\}$ is

$$P(O|Q, \omega) = \prod_{t=1}^{T} b_{q_t}(o_t) \tag{15}$$

Then, the joint probability distribution of O and Q is,

$$P(O, Q|\omega) = \pi_{q_1} b_{q_1}(o_1) \prod_{t=1}^{T-1} a_{q_t q_{t+1}} b_{q_{t+1}}(o_{t+1}) \tag{16}$$

Finally, sum all the Q, we can get the probability of specifical vehicle trajectory observation under the model $\omega = (A, B, \pi)$.

$$P(O|\omega) = \sum_{q_1, q_2, q_3, \dots, q_T} \pi_{q_1} b_{q_1}(o_1) \prod_{t=1}^{T-1} a_{q_t q_{t+1}} b_{q_{t+1}}(o_{t+1}) \tag{17}$$

Nevertheless, the calculation of the possibility in Equation (17) is relatively computationally expensive due to its high time complexity ($O(TN^T)$). Therefore, we used the forward and backward algorithm to solve this problem [68]–[70]. The algorithm can calculate the probability under the model by a given sequence of observations at a low complexity ($O(N^2T)$).

We define two variables $\alpha_t(i)$ and $\beta_t(i)$. $\alpha_t(i)$ called the forward parameter that means the probability of past observations in a given state q_t at time t. $\beta_t(i)$ represents back parameter that means the probability of the future observations in a given state q_t at time t. A schematic diagram of the forward and back parameters is presented in Fig. 6, while Fig. 7 shows the computation process of the joint event using the forward and back algorithm.

This algorithm can calculate the probability that a model generated a sequence of observations. For a known $\omega = (A, B, \pi)$, at time t, if the observation series is $o_1, o_2, o_3, \dots, o_t$, the forward possibility is

$$\alpha_t(i) = P(o_1, o_2, o_3, \dots, o_t, q_t = S_i|\omega) \tag{18}$$

Then, the forward formula can be represented by the base case $\alpha_1(i)$ and inductive step. Following the definition of the parameters of the HMMs, debase case $\alpha_1(i)$ is

$$\alpha_1(i) = \pi_i b_i(o_1), \forall i = 1,2, \dots, N \tag{19}$$

Sum of all the different probabilities of getting to state j times the emission. The inductive step is

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(o_{t+1}),$$
$$\forall j = 1,2, \dots, N, \forall t = 1,2, \dots, T-1 \tag{20}$$

Then, the final step is that sum all the probabilities which will end up the state given the observation sequence. The probability of the observation O by given the model ω is

$$P(O|\omega) = \sum_{i=1}^{N} \alpha_T(i) \tag{21}$$

For the backward possibility, at time t, the observation is $O_w = (o_{t+1}, o_{t+2}, \dots, o_T)$. The backward algorithm is

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, \dots, o_T|q_t = S_i, \omega) \tag{22}$$

The base case and inductive step are

$$\beta_T(i) = 1, \forall i = 1,2, \dots, N \tag{23}$$

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij}\, b_j(o_{t+1}) \beta_{t+1}(j),$$
$$\forall i = 1,2, \dots, N, \forall t = T-1, T-2, \dots, 1 \tag{24}$$

The probability of the observation O by given the model ω is

$$P(O|\omega) = \sum_{i=1}^{N} \pi_j b_j(o_1) \beta_1(j) \tag{25}$$

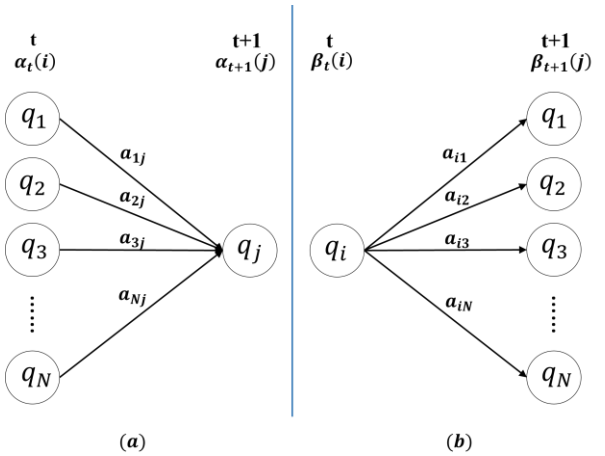


Fig. 6. Forward and backward algorithm in HMM-4-C. (a) forward parameter. (b) backward parameter.

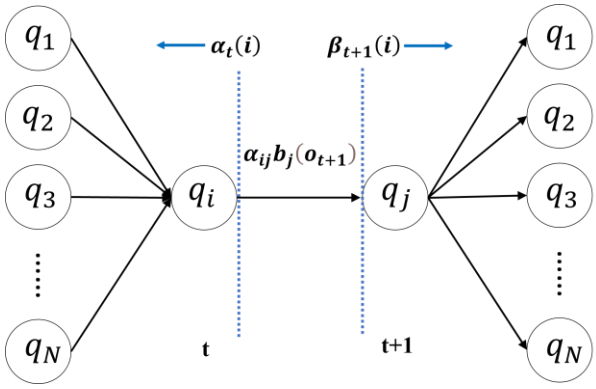


Fig. 7. The computation process of the joint event.

D. EM algorithm (Baum-Welch algorithm)

In this research, the basic parameters $\omega = (A, B, \pi)$ are not given. We need to use the observed trajectory datasets to estimate the parameters. Here, we used Expectation-Maximization (EM) algorithm [68], [71], [72] to estimate $\omega = (A, B, \pi)$. In the other words, for a given set of trajectory observation, we need to find parameters $\omega = (A, B, \pi)$ to maximize the likelihood of generating that the sequence of observation. Therefore, in this section, we estimate three different parameters ω_1, ω_2 and ω_3 for the three different scenarios using the EM algorithm.

In each model, we first define initial set of parameters $\omega_0 = (A_0, B_0, \pi_0)$. The objective is to find the optimal set of parameters $\bar{\omega} = (\bar{A}, \bar{B}, \bar{\pi})$ that maximize our observations from training datasets.

$$\omega_0 = (A_0, B_0, \pi_0) \quad (26)$$

$$\bar{\omega} = (\bar{A}, \bar{B}, \bar{\pi}) \quad (27)$$

Therefore, we resort to looping the expectation step (E-step) and the maximization step (M-step) in EM algorithm to update the parameters $\omega = (A, B, \pi)$ [72]–[75]. Following the EM algorithm, we evaluate the expectation of the complete data log-likelihood function, under the posterior distribution of hidden states based on the current estimate of the parameters ω, the likelihood function of E step is

$$Q(\omega, \bar{\omega}) = \sum_I P(O, Q|\bar{\omega}) \log P(O, Q|\omega) \quad (28)$$

Depend on Equation (14)-(17), the likelihood function be expressed as

$$Q(\omega, \bar{\omega}) = \sum_I \left(\log \pi_{i_1}\right) P(O, Q|\bar{\omega}) + \sum_I \left(\sum_{t=1}^{T-1} \log a_{i_t i_{t+1}}\right) P(O, Q|\bar{\omega}) + \sum_I \left(\sum_{t=1}^{T} \log b_{i_t}(o^t)\right) P(O, Q|\bar{\omega}) \quad (24)$$

Where T is the length of O and o^t is the t th observation.

The analytical solution for maximize the likelihood function given by Equation (24) involves setting the derivatives to zero subject to the following constraints.

$$\sum_{j=1}^{N} \alpha_{ij} = 1 \quad (25)$$

$$\sum_{k=1}^{M} b_j(k) = 1 \quad (26)$$

$$\sum_{i=1}^{N} \pi_i = 1 \quad (27)$$

Therefore, the final formula $\omega^{\langle new \rangle} = \left(A^{\langle new \rangle}, B^{\langle new \rangle}, \pi^{\langle new \rangle}\right)$ is

$$a_{ij}{}^{\langle new \rangle} = \frac{\sum_{t=1}^{T-1} P\left(O, q_t = S_i, q_{t+1} = S_j | \omega^{\langle old \rangle}\right)}{\sum_{t=1}^{T-1} P\left(O, q_t = S_i | \omega^{\langle old \rangle}\right)} \quad (28)$$

$$b_j(k)^{\langle new \rangle} = \frac{\sum_{t=1}^{T} P\left(O, q_t = S_i | \omega^{\langle old \rangle}\right) Q(o_t = VT_k)}{\sum_{t=1}^{T} P\left(O, q_t = S_i | \omega^{\langle old \rangle}\right)} \quad (29)$$

$$\pi_i{}^{\langle new \rangle} = \frac{P\left(O, q_1 = S_i | \omega^{\langle old \rangle}\right)}{P\left(O | \omega^{\langle old \rangle}\right)} \quad (30)$$

In the M step, we obtain a new estimate of the parameters $\bar{\omega}^{\langle new \rangle}$ by maximizing $Q(\omega, \bar{\omega})$ subject to the constraints, which is,

$$\bar{\omega}^{\langle new \rangle} \leftarrow \underset{\omega}{\operatorname{argmax}}\, Q(\omega, \bar{\omega}) \quad (31)$$

We then loop the algorithm until it finds the $\bar{\omega}$ that maximizes the log-likelihood function, and generate the new $\bar{\omega} = (\bar{A}, \bar{B}, \bar{\pi})$, which indicates a new model. Algorithm 1 provides the process of the EM algorithm. Figs. 8 and 9 provide the steps of the EM algorithm and detection method in this research.

Algorithm 1: EM algorithm

Input: Observation sequence $\boldsymbol{O}$, Initial $\boldsymbol{\omega_0} = (\boldsymbol{A_0}, \boldsymbol{B_0}, \boldsymbol{\pi_0})$

Output: Updated parameters of HMM-4-C, $\bar{\boldsymbol{\omega}} = (\bar{\boldsymbol{A}}, \bar{\boldsymbol{B}}, \bar{\boldsymbol{\pi}})$

While until convergence **do**

$$\boldsymbol{a_{ij}}^{\langle new \rangle} = \frac{\sum_{t=1}^{T-1} \boldsymbol{P}\left(\boldsymbol{O}, \boldsymbol{q_t} = \boldsymbol{S_i}, \boldsymbol{q_{t+1}} = \boldsymbol{S_j} \middle| \boldsymbol{\omega}^{\langle old \rangle}\right)}{\sum_{t=1}^{T-1} \boldsymbol{P}\left(\boldsymbol{O}, \boldsymbol{q_t} = \boldsymbol{S_i} \middle| \boldsymbol{\omega}^{\langle old \rangle}\right)}$$

$$\boldsymbol{b_j(k)}^{\langle new \rangle} = \frac{\sum_{t=1}^{T} \boldsymbol{P}\left(\boldsymbol{O}, \boldsymbol{q_t} = \boldsymbol{S_i} \middle| \boldsymbol{\omega}^{\langle old \rangle}\right) \boldsymbol{Q}(\boldsymbol{o_t} = \boldsymbol{VT_k})}{\sum_{t=1}^{T} \boldsymbol{P}\left(\boldsymbol{O}, \boldsymbol{q_t} = \boldsymbol{S_i} \middle| \boldsymbol{\omega}^{\langle old \rangle}\right)}$$

$$\boldsymbol{\pi_i}^{\langle new \rangle} = \frac{\boldsymbol{P}\left(\boldsymbol{O}, \boldsymbol{q_1} = \boldsymbol{S_i} \middle| \boldsymbol{\omega}^{\langle old \rangle}\right)}{\boldsymbol{P}\left(\boldsymbol{O} \middle| \boldsymbol{\omega}^{\langle old \rangle}\right)}$$

return $\bar{\boldsymbol{\omega}} = (\bar{\boldsymbol{A}}, \bar{\boldsymbol{B}}, \bar{\boldsymbol{\pi}})$

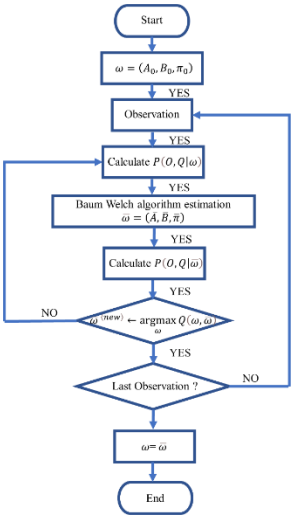


Fig. 8. The EM algorithm.

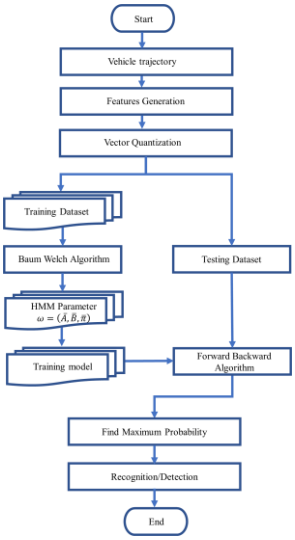


Fig. 9. The heuristic approach of modeling vehicle trajectories.

E. Benchmark Method for Comparison: Bidirectional Long Short-Term Memory (Bi-LSTM)

To evaluate HMM-4-C proposed in the research, a commonly used neural network known as Bidirectional Long Short-Term Memory (Bi-LSTM) was used to detect the trajectories. Bi-LSTM considers both past and future data in its learnings, enhances its learning ability than LTSM. Detailed model of Bi-LSTM can be found in the following literature [76]–[78]. To evaluate the performance of HMM-4-C, same trajectory datasets were also modeled using the Bi-LSTM. To prepare the vehicle trajectory datasets for analysis, the vehicle trajectory datasets were also divided into the same training and testing datasets. For each scenario, we obtain third-two trajectory datasets, out of which twenty were randomly selected as the training datasets. The remaining twelve datasets were used for testing the performance of the models. The trajectory datasets were modeled using the Bi-LSTM and compared with HMM-4-C in the next section.

F. Evaluation of the model's performance

Recall and precision are commonly used in the fields of machine learning, particularly in classification and detection tasks [79]–[81]. Precision assesses how many of the items identified as positive by the model are actually positive. Recall measures how many of the actual positives the model captures by labeling them as positive. Often, there's a trade-off between recall and precision. Improving recall may reduce precision and vice versa. To consider both precision and recall, one can use the F-measure score, also named F1 score was used to both consider both precision and recall. F-measure score usually is used on evaluation model' s the classification problem. Specifically, F1 score is defined by the results of precision and recall. Recall and precision are defined as Equation 32-34.

$$Precision = \frac{TP}{TP + FP} \tag{32}$$

$$Recall = \frac{TP}{TP + FN} \tag{33}$$

$$F1\ score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{34}$$

Where TP is True Positive numbers, FP is False Positive numbers and FN is False Negative numbers.

G. Training and Testing Datasets

To prepare the vehicle trajectory datasets for modeling, the vehicle trajectory datasets were divided into two parts: the training and testing datasets. We first collected 96 trajectory datasets from the 32 participants who drove the designated scenario three times. For each scenario, we obtain 32 trajectory datasets, out of which 20 were randomly selected as the training datasets. Three different models were then trained using corresponding training datasets. The remaining 12 trajectories in each scenario were used as testing datasets to evaluate the performance of HMM-4-C.

In the testing process, we tested 36 testing datasets, including 12 datasets from each scenario, using the three trained models. A forward and backward algorithm was used to calculate the probability of each testing datasets under different models. To determine the detection result for a testing dataset, the corresponding scenario

with the maximum probability was considered as the detection result. Hundreds of trainings and tests were performed with different random seeds.

In this research, we use a range of detection as 200ft, following the guidelines in the *Detector Locations for Conventional Traffic Systems in Traffic Control Systems Handbook* [82]. It means all trajectories within the range of 200ft close to the traffic signal were recorded for detection. We assume that sensors can fully capture the vehicle trajectories without errors.

All the detection results, including true positive (TP), false negative (FN), false positive (FP), and true negative (TN) were recorded for creating the confusion matrix.

Algorithm 2: HMM-4-C Detection

Input: Testing datasets;
Output: detection results;
for $\boldsymbol{i}$ = each testing dataset $\boldsymbol{VT}$ **do**
  for $\boldsymbol{j}$ = each $\boldsymbol{Model}$ **do**
    //calculate the probability of specific test sequences under different models
    Prob ($\boldsymbol{j}$) = Forward and backward Algorithm ($\boldsymbol{Model}(\boldsymbol{j}), \boldsymbol{VT}(\boldsymbol{i})$)
    Prob of $\boldsymbol{VT}(\boldsymbol{i})$ = Append (Prob ($\boldsymbol{j}$))
  end for
  //choose the maximum probability
  Most likely = maximum (Prob of VT ($\boldsymbol{i}$))
  detection/recognition = index (maximum (Prob of VT ($\boldsymbol{i}$)))
end for
return detection results;

IV. RESULTS

A. Simulator Experiment Result: Trajectory Analysis

The driving simulator experiment has resulted in drivers' trajectory data when they drive through the three expriment scenarios, namely, "pass", "stop", and "cyberattack". The time distance diagrams for the three scenarios are presented in Fig. 10.

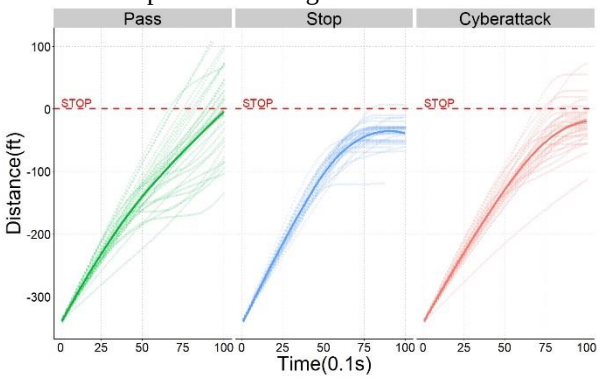
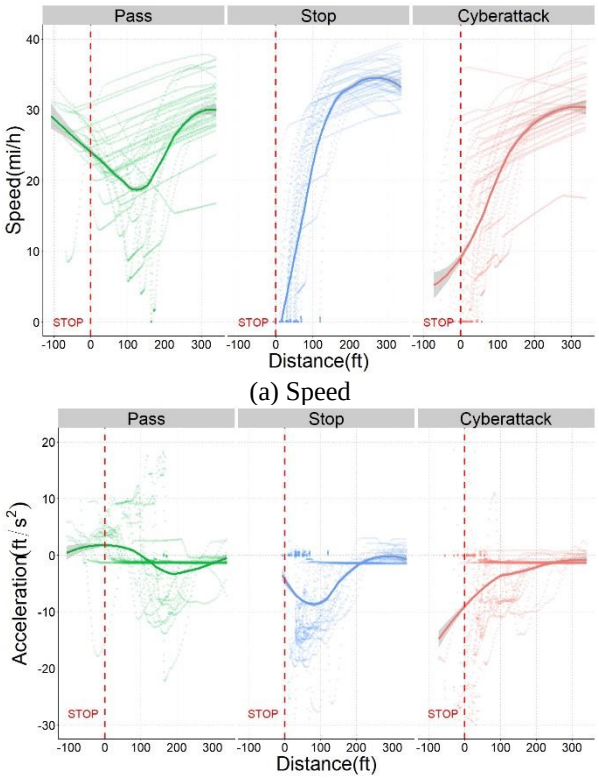


Fig. 10. The relationships between time and distance in three different scenarios.

A total of thirty-two vehicle trajectories were combined to generate a loess regression curve. The time 0 denotes the moment at which the RL countdown message was displayed to the drivers. The distance 0 indicates the location of the stop line.



(a) Speed

(b) Acceleration

Fig. 11. The feature of speed and acceleration with distance under three scenarios.

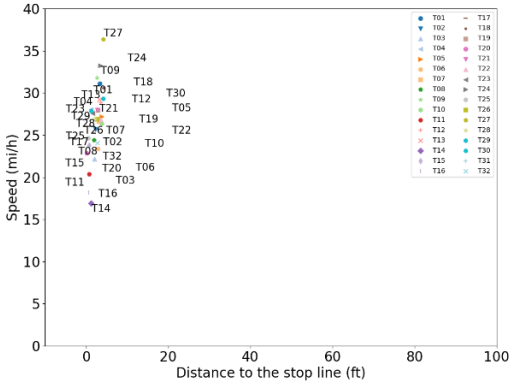
In the Pass scenario, participants faced a short RL countdown time. The figure indicates that most of the participants drove through the intersection at a consistent speed. A few participants slowed down with little uncertainty when facing the red light before approaching the intersection. In the Stop scenario, participants faced a long-time RL and had to come to a complete stop before getting into the intersection. Most participants stopped a distance away from the stop line. Only some participants stopped before the line in the Cyberattack scenario. It means eleven out of thirty-two participants drove into the intersection and caused severe car accidents. The participants had not realized the red light was still on after the RL countdown ended, and the vehicle kept approaching the intersection. As a result, facing a cyberattack changed driving behaviors and vehicle trajectories at the intersection. These unusual driving behaviors provided unique features for detection.

Fig. 11 depict the speed and acceleration diagram for each scenario. Fig. 12 shows the last speed value before the stop line. In the Pass scenario, fifteen drivers maintained a stable deceleration rate to pass the intersection, showing the straight speed lines in Fig. 11b.

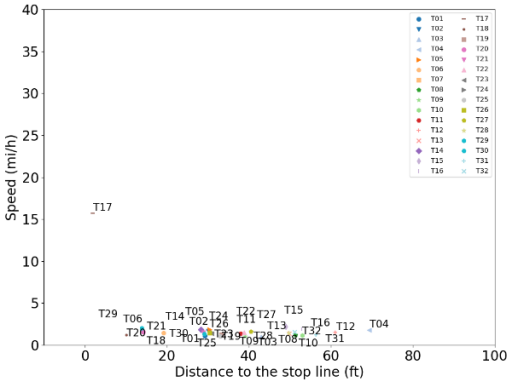
It indicates that these drivers did not use the brake or gas pedal during that time but kept a uniform deceleration rate. The rest of the participants slowed their speed when they approached the intersection, even though they had an short-time RL countdown. When the countdown ended, the light turned green. They had to increase their speed to cross the intersection at the distance of 100ft. Eventually, speed curves in Fig. 11a show the "V" type. The "V" type speed curves in the Pass scenario in Fig. 10a correspond to acceleration curves in Fig. 11b. The acceleration curves show a decrease in acceleration to a negative value of 10 ft/s^2 followed by an increase to a positive value of 10 ft/s^2 to accelerate and pass through the intersection.

In the Stop scenario, only one participant stopped the vehicle exceeding the stop line, and the rest stopped before the stop line. Participants continuously used brakes and reduced the speed when they approached the intersection until the speed to 0. Most participants began using the brakes at around 200ft, and the deceleration rate generally came to zero when the vehicle came to a stop.

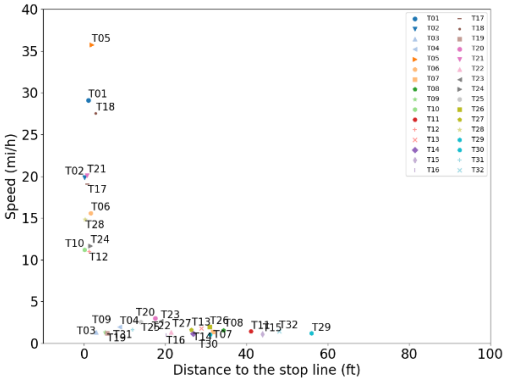
In the Cyberattack scenario, eleven participants entered the intersection at a relatively high speed, shown in Fig. 12c. These participants were attacked by the cybercriminals successfully. The rest of the participants used the brake in a short time to prevent getting into the intersection. The deceleration curves show that participants applied a high value close to negative 30ft/s^2. The hard brake action in the Cyberattack scenario suggests the unexpected events that occurred at that time. The experiment results showed more than one third of participants were attacked and continued to enter the intersection when the red light was on, resulting in serious accidents. The rest of participants attempted to stop by applying the brakes abruptly, resulting in a significant and characteristic deceleration rate values.



(a) Pass



(b) Stop



(c) Cyberattack

Fig. 12. Speed at the final moment before the stop line under three scenarios.

B. Detection and modeling

The estimated transition matrix of HMM-4-C is shown in Fig. 13.

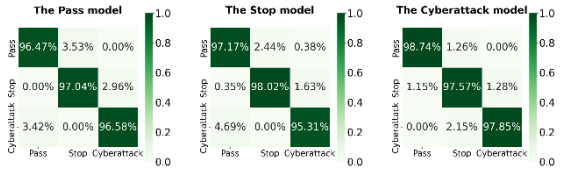


Fig. 13. The estimated transition matrix of HMM-4-C.

A high transition probability of the position a_{11}, a_{22}, a_{33} means the current state model is not easily changed to another and has higher accuracy for detecting specific trajectory features. Notably, the Cyberattack model has higher diagonal probabilities than the other two models. This is because the features of driving behaviors and vehicle trajectories under the Cyberattack scenarios are more distinctive than other scenarios, as can be inferred from Fig. 11. In the Bi-LSTM modeling process, we performed a hundred epochs for training. The training accuracy and loss value of training process are shown in

Fig. 14. The final confusion matrixes of two models are shown in Fig. 15.

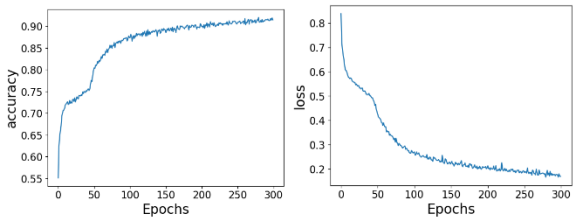


Fig. 14. The training accuracy and loss of the Bi-LSTM.

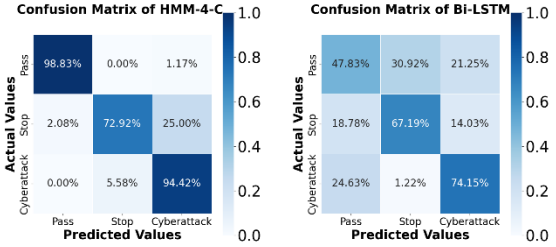


Fig. 15. Final Detection results.

In Fig. 15, HMM-4-C showed higher detection accuracy for all three scenarios, with the Pass scenario achieving 98.83% true positive rate and the Stop and Cyberattack scenarios achieving 72.92% and 94.42% true positive rates, respectively. However, there were some misdetections, with 5.58% of the Cyberattack testing datasets being classified as Stop. 25% of the Stop testing datasets were classified as Cyberattack. In addition, the results from the confusion matrix of Bi-LSTM show a relative lower accuracy than HMM-4-C. The Pass scenario achieving 47.83% true positive rate and the Stop and Cyberattack scenarios achieving 67.19% and 74.15% true positive rates, respectively. The detection results show that 94.42% of cyberattack trajectories were successfully detected using HMM-4-C. HMM-4-C performed better than the Bi-LSTM in detecting cyberattacked trajectories in our research.

The results of precision, recall and F1 score are presented in Fig. 16.

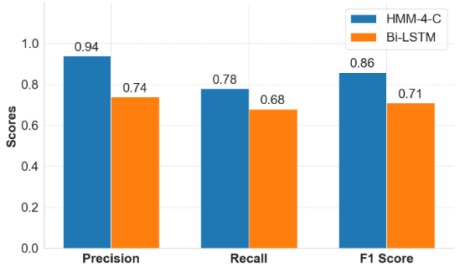


Fig. 16. The comparison of cyberattack detection results in the two models.

In the term of cyberattack detection results, HMM-4-C correctly predicted positive instances 94% of the time when it claimed something was positive, while the Bi-LSTM model was correct 74% of the time. The application of detection cyberattack events always require high precision, HMM-4-C would be preferable based on the provided results in the context. Moreover, HMM-4-C correctly identified 78% of the actual positive events, while the Bi-LSTM model identified 68% of them.

Given the F1 scores of the two models, it's evident that HMM-4-C outperformed the Bi-LSTM on the cyberattack datasets, with an F1 score of 0.86 versus 0.71, respectively. These indicates the high detection results achieved by the model demonstrates the potential of using HMM-4-C and trajectory data for detecting anomalous driving behaviors, which can contribute to the development of more robust and secure transportation systems.

V. DISCUSSIONS

From the perspective of detection results, HMM-4-C achieves better outcomes than Bi-LSTM. In many areas, Bi-LSTM and other deep learning methods have outperformed HMMs [83]–[85]. However, in this research, the higher detection accuracy is attributed to the characteristics features under the cyberattack scenario. One reason is that HMMs are inherently designed for sequences and can effectively capture short-term temporal dependencies between states [86], [87]. HMMs based methods are based on a probabilistic framework which provides a natural way to handle uncertainties in vehicle trajectories. Since cyberattacks occur abruptly and within seconds and cyberattacked trajectory change in frequently and abruptly, the participants react by applying the brakes suddenly, as shown in Figs 10 and 11. This leads to significant and characteristic state changes over a short period. The trajectory patterns for detecting cyberattacks are mostly local (short sequences) [88]. Therefore, HMM-4-C is better suited than the deep learning methods, which often captures longer dependencies.

In addition, HMMs can work effectively with smaller datasets than other deep learning methods [89]. In our research, training HMM-4-C requires iterative procedures i.e., the EM algorithm, which only converge with 20 examples. As mentioned by previous literature, HMMs outperform Bi-LSTMs when the training dataset is insufficient [90]. Most deep learning models, like Bi-LSTMs, typically require larger datasets to train effectively and generalize well [91]. The intricate architecture allows them to model long-term dependencies and complex relationships in the data. This capability is a double-edged sword: while it lets the model capture sophisticated patterns, it also demands more data to prevent overfitting and to train effectively.

For cyberattacks with smaller dataset sizes of vehicle trajectories, HMMs are more suitable in current situation.

The errors in the detection results of HMM-4-C shown in Fig. 15, primarily arise from the relative similarities in trajectory features between the cyberattack and stop scenarios. These similarities are more pronounced than those with the pass situation, as shown in Figs. 10 and 11. This can lead to confusion in the classification process. According to the aforementioned trajectory diagrams, people tend to stop when facing a cyberattack. The misdetection arises because the features of cyberattack trajectories are more similar to the stop scenario than to the pass scenario.

VI. CONCLUSIONS

This study introduces a novel HMMs based method called HMM-4-C for detecting cyberattacks at connected signalized intersections. The findings validate that Markov Chain-based detection methods have an edge over the deep learning methods in detecting cyberattacks. The advantage stems from the specific short-term temporal dependency of cyberattack moment and the dataset size related to the current state of cyberattack events.

This current study has following limitations. The training and testing datasets were based on driving simulator. Future research will take a further step by involving field road driving to validate the findings. In addition, this research also employed a fixed range of detection boundaries based on the location information. Future research will consider dynamic or multi-scale perception and detection for further improved accuracy and efficiency. Also, we will implement the advanced hybrid detection model, incorporating data fusion to focus on enhancing accuracy.

In conclusion, this method provides a proof of the concept from a newer angle to achieve cyberattack detection from the physical perspective in the cyber-physical system instead of addressing the issue from the traditional sensor intrusions and network anomalies perspective. The research has the potential to predict and proactively detect future cyberattacks at connected signalized intersections. It can be used in coordination with cyber-based methods to further confirm the occurrences of cyberattacks in the early stages.

VI. REFERENCES

[1] J. F. Powers, *Cyber terrorism and extremism as threat to critical infrastructure protection*. Ljubljana; Tampa: Ministry of Defense, Republic of Slovenia : Institute for Corporative Security Studies : Joint Special Operations University, 2020. Accessed: Dec. 21, 2021. [Online]. Available: https://dk.mors.si/IzpisGradiva.php?id=1121

[2] E. S. Canepa and C. G. Claudel, "Spoofing cyber attack detection in probe-based traffic monitoring systems using mixed integer linear programming," in *2013 International Conference on Computing, Networking and Communications (ICNC)*, San Diego, CA: IEEE, Jan. 2013, pp. 327–333. doi: 10.1109/ICCNC.2013.6504104.

[3] S. K. Khan, N. Shiwakoti, P. Stasinopoulos, and Y. Chen, "Cyber-attacks in the next-generation cars, mitigation techniques, anticipated readiness and future directions," *Accident Analysis & Prevention*, vol. 148, p. 105837, Dec. 2020, doi: 10.1016/j.aap.2020.105837.

[4] B. Sheehan, F. Murphy, M. Mullins, and C. Ryan, "Connected and autonomous vehicles: A cyber-risk classification framework," *Transportation Research Part A: Policy and Practice*, vol. 124, pp. 523–536, Jun. 2019, doi: 10.1016/j.tra.2018.06.033.

[5] M. Gupta, J. Benson, F. Patwa, and R. Sandhu, "Secure V2V and V2I Communication in Intelligent Transportation Using Cloudlets," *IEEE Transactions on Services Computing*, vol. 15, no. 4, pp. 1912–1925, Jul. 2022, doi: 10.1109/TSC.2020.3025993.

[6] S. E. Huang, Y. Feng, and H. X. Liu, "A data-driven method for falsified vehicle trajectory identification by anomaly detection," *Transportation Research Part C: Emerging Technologies*, vol. 128, p. 103196, Jul. 2021, doi: 10.1016/j.trc.2021.103196.

[7] S. Jung, J. Kim, M. Levorato, C. Cordeiro, and J.-H. Kim, "Infrastructure-Assisted On-Driving Experience Sharing for Millimeter-Wave Connected Vehicles," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 8, pp. 7307–7321, Aug. 2021, doi: 10.1109/TVT.2021.3094806.

[8] A. P. Mdee, M. T. R. Khan, J. Seo, and D. Kim, "Security Compliant and Cooperative Pseudonyms Swapping for Location Privacy Preservation in VANETs," *IEEE Transactions on Vehicular Technology*, pp. 1–15, 2023, doi: 10.1109/TVT.2023.3254660.

[9] I. Butun, P. Österberg, and H. Song, "Security of the Internet of Things: Vulnerabilities, Attacks and Countermeasures," *IEEE Commun. Surv. Tutorials*, vol. 22, no. 1, pp. 616–644, 2020, doi: 10.1109/COMST.2019.2953364.

[10] M. Chowdhury, M. Islam, and Z. Khan, "Security of Connected and Automated Vehicles," *arXiv preprint arXiv:2012.13464*, 2020.

[11] J. Haddad and B. Mirkin, "Resilient perimeter control of macroscopic fundamental diagram networks under cyberattacks," *Transportation Research Part B: Methodological*, vol. 132, pp. 44–59, Feb. 2020, doi: 10.1016/j.trb.2019.01.020.

[12] J. Petit and S. E. Shladover, "Potential Cyberattacks on Automated Vehicles," *IEEE Trans. Intell. Transport. Syst.*, pp. 1–11, 2014, doi: 10.1109/TITS.2014.2342271.

[13] S. Grad, "Engineers who hacked into L.A. traffic signal computer, jamming streets, sentenced," *LA Times Blogs - L.A. NOW*, Dec. 01, 2009. https://latimesblogs.latimes.com/lanow/2009/12/engineers-who-hacked-in-la-traffic-signal-computers-jamming-traffic-sentenced.html (accessed Apr. 17, 2022).

[14] M. Prigg, "New York's traffic lights HACKED," *Mail Online*, Apr. 30, 2014.

https://www.dailymail.co.uk/sciencetech/article-2617228/New-Yorks-traffic-lights-HACKED-technique-work-world.html (accessed Apr. 17, 2022).

[15] J. R. Miller, "Hackers Crack Into Texas Road Sign, Warn of Zombies Ahead," *Fox News*, Mar. 25, 2015. https://www.foxnews.com/story/hackers-crack-into-texas-road-sign-warn-of-zombies-ahead (accessed Apr. 17, 2022).

[16] D. Morris, G. Madzudzo, and A. G. Perez, "Cybersecurity and the auto industry: the growing challenges presented by connected cars," *IJATM*, vol. 18, no. 2, p. 105, 2018, doi: 10.1504/IJATM.2018.092187.

[17] M. T. Whitty, "Is There a Scam for Everyone? Psychologically Profiling Cyberscam Victims," *Eur J Crim Policy Res*, vol. 26, no. 3, pp. 399–409, Sep. 2020, doi: 10.1007/s10610-020-09458-z.

[18] Y. Benmessaoud, L. Cherrat, and M. Ezziyyani, "Real-Time Self-Adaptive Traffic Management System for Optimal Vehicular Navigation in Modern Cities," *Computers*, vol. 12, no. 4, Art. no. 4, Apr. 2023, doi: 10.3390/computers12040080.

[19] A. M. de Souza, N. L. S. da Fonseca, and L. Villas, "A fully-distributed advanced traffic management system based on opportunistic content sharing," in *2017 IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–6. doi: 10.1109/ICC.2017.7997071.

[20] F. van Wyk, Y. Wang, A. Khojandi, and N. Masoud, "Real-Time Sensor Anomaly Detection and Identification in Automated Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 1264–1276, Mar. 2020, doi: 10.1109/TITS.2019.2906038.

[21] M. N. Mejri, N. Achir, and M. Hamdi, "A new security games based reaction algorithm against DOS attacks in VANETs," in *2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, Jan. 2016, pp. 837–840. doi: 10.1109/CCNC.2016.7444896.

[22] M. S. Faughnan *et al.*, "Risk analysis of Unmanned Aerial Vehicle hijacking and methods of its detection," in *2013 IEEE Systems and Information Engineering Design Symposium*, Apr. 2013, pp. 145–150. doi: 10.1109/SIEDS.2013.6549509.

[23] L. Karim and A. Boulmakoul, "Trajectory-based Modeling for Fraud Detection and Analytics: Foundation and Design," in *2021 IEEE/ACS 18th International Conference on Computer Systems and Applications (AICCSA)*, Tangier, Morocco: IEEE, Nov. 2021, pp. 1–7. doi: 10.1109/AICCSA53542.2021.9686920.

[24] K. Kumaran Santhosh, D. P. Dogra, P. P. Roy, and A. Mitra, "Vehicular Trajectory Classification and Traffic Anomaly Detection in Videos Using a Hybrid CNN-VAE Architecture," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 11891–11902, Aug. 2022, doi: 10.1109/TITS.2021.3108504.

[25] I. Kalinov *et al.*, "WareVision: CNN Barcode Detection-Based UAV Trajectory Optimization for Autonomous Warehouse Stocktaking," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6647–6653, Oct. 2020, doi: 10.1109/LRA.2020.3010733.

[26] Y. Liu, K. Zhao, G. Cong, and Z. Bao, "Online Anomalous Trajectory Detection with Deep Generative Sequence Modeling," in *2020 IEEE 36th International Conference on Data Engineering (ICDE)*, Apr. 2020, pp. 949–960. doi: 10.1109/ICDE48307.2020.00087.

[27] C. Zhang, Z. Ni, and C. Berger, "Spatial-Temporal-Spectral LSTM: A Transferable Model for Pedestrian Trajectory Prediction," *IEEE Transactions on Intelligent Vehicles*, pp. 1–14, 2023, doi: 10.1109/TIV.2023.3285804.

[28] T. Hickling, N. Aouf, and P. Spencer, "Robust Adversarial Attacks Detection based on Explainable Deep Reinforcement Learning for UAV Guidance and Planning," *IEEE Transactions on Intelligent Vehicles*, pp. 1–14, 2023, doi: 10.1109/TIV.2023.3296227.

[29] Y. Xue and W. Chen, "Multi-Agent Deep Reinforcement Learning for UAVs Navigation in Unknown Complex Environment," *IEEE Transactions on Intelligent Vehicles*, pp. 1–14, 2023, doi: 10.1109/TIV.2023.3298292.

[30] N. Lin, C. Zong, M. Tomizuka, P. Song, Z. Zhang, and G. Li, "An Overview on Study of Identification of Driver Behavior Characteristics for Automotive Control," *Mathematical Problems in Engineering*, vol. 2014, pp. 1–15, 2014, doi: 10.1155/2014/569109.

[31] C.-E. Wu, W.-Y. Yang, H.-C. Ting, and J.-S. Wang, "Traffic pattern modeling, trajectory classification and vehicle tracking within urban intersections," in *2017 International Smart Cities Conference (ISC2)*, Sep. 2017, pp. 1–6. doi: 10.1109/ISC2.2017.8090791.

[32] Y. Zhao, S. Shen, and H. X. Liu, "A hidden Markov model for the estimation of correlated queues in probe vehicle environments," *Transportation Research Part C: Emerging Technologies*, vol. 128, p. 103128, Jul. 2021, doi: 10.1016/j.trc.2021.103128.

[33] Z.-Q. Liu, X. Ge, Q.-L. Han, Y.-L. Wang, and X.-M. Zhang, "Secure Cooperative Path Following of Autonomous Surface Vehicles Under Cyber and Physical Attacks," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 6, pp. 3680–3691, Jun. 2023, doi: 10.1109/TIV.2023.3270266.

[34] P. Wang, G. Yu, X. Wu, H. Qin, and Y. Wang, "An extended car-following model to describe connected traffic dynamics under cyberattacks," *Physica A: Statistical Mechanics and its Applications*, vol. 496, pp. 351–370, Apr. 2018, doi: 10.1016/j.physa.2017.12.013.

[35] L. Cui, J. Hu, B. B. Park, and P. Bujanovic, "Development of a simulation platform for safety impact analysis considering vehicle dynamics, sensor errors, and communication latencies: Assessing cooperative adaptive cruise control under cyber attack," *Transportation Research Part C: Emerging Technologies*, vol. 97, pp. 1–22, Dec. 2018, doi: 10.1016/j.trc.2018.10.005.

[36] Q. A. Chen, Y. Yin, Y. Feng, Z. M. Mao, and H. X. Liu, "Exposing Congestion Attack on Emerging Connected Vehicle based Traffic Signal Control," in *Proceedings 2018 Network and Distributed System Security Symposium*, San Diego, CA: Internet Society, 2018. doi: 10.14722/ndss.2018.23222.

[37] Á. Török, Z. Szalay, G. Uti, and B. Verebélyi, "Modelling the effects of certain cyber-attack methods

on urban autonomous transport systems, case study of Budapest," *J Ambient Intell Human Comput*, vol. 11, no. 4, pp. 1629–1643, Apr. 2020, doi: 10.1007/s12652-019-01264-8.

[38] M. Amoozadeh *et al.*, "Security vulnerabilities of connected vehicle streams and their impact on cooperative driving," *IEEE Commun. Mag.*, vol. 53, no. 6, pp. 126–132, Jun. 2015, doi: 10.1109/MCOM.2015.7120028.

[39] S. F. Meyer, R. Elvik, and E. Johnsson, "Risk analysis for forecasting cyberattacks against connected and autonomous vehicles," *J Transp Secur*, vol. 14, no. 3–4, pp. 227–247, Dec. 2021, doi: 10.1007/s12198-021-00236-4.

[40] Y. Li, Y. Tu, Q. Fan, C. Dong, and W. Wang, "Influence of cyber-attacks on longitudinal safety of connected and automated vehicles," *Accident Analysis & Prevention*, vol. 121, pp. 148–156, Dec. 2018, doi: 10.1016/j.aap.2018.09.016.

[41] P. Wang, X. Wu, and X. He, "Modeling and analyzing cyberattack effects on connected automated vehicular platoons," *Transportation Research Part C: Emerging Technologies*, vol. 115, p. 102625, Jun. 2020, doi: 10.1016/j.trc.2020.102625.

[42] Q. He, X. Meng, and R. Qu, "Towards a Severity Assessment Method for Potential Cyber Attacks to Connected and Autonomous Vehicles," *Journal of Advanced Transportation*, vol. 2020, pp. 1–15, Sep. 2020, doi: 10.1155/2020/6873273.

[43] Z. H. Khattak, B. L. Smith, and M. D. Fontaine, "Impact of cyberattacks on safety and stability of connected and automated vehicle platoons under lane changes," *Accident Analysis & Prevention*, vol. 150, p. 105861, Feb. 2021, doi: 10.1016/j.aap.2020.105861.

[44] Y. Feng, S. Huang, Q. A. Chen, H. X. Liu, and Z. M. Mao, "Vulnerability of Traffic Control System Under Cyberattacks with Falsified Data," *Transportation Research Record*, vol. 2672, no. 1, pp. 1–11, Dec. 2018, doi: 10.1177/0361198118756885.

[45] K. A. Perrine, M. W. Levin, C. N. Yahia, M. Duell, and S. D. Boyles, "Implications of traffic signal cybersecurity on potential deliberate traffic disruptions," *Transportation Research Part A: Policy and Practice*, vol. 120, pp. 58–70, Feb. 2019, doi: 10.1016/j.tra.2018.12.009.

[46] B. DeBruhl, S. Weerakkody, B. Sinopoli, and P. Tague, "Is your commute driving you crazy?: a study of misbehavior in vehicular platoons," in *Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, New York New York: ACM, Jun. 2015, pp. 1–11. doi: 10.1145/2766498.2766505.

[47] W. Li and H. Song, "ART: An Attack-Resistant Trust Management Scheme for Securing Vehicular Ad Hoc Networks," *IEEE Trans. Intell. Transport. Syst.*, vol. 17, no. 4, pp. 960–969, Apr. 2016, doi: 10.1109/TITS.2015.2494017.

[48] Z. Abdollahi Biron, S. Dey, and P. Pisu, "Real-Time Detection and Estimation of Denial of Service Attack in Connected Vehicle Systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 12, pp. 3893–3902, Dec. 2018, doi: 10.1109/TITS.2018.2791484.

[49] E. Mousavinejad, F. Yang, Q.-L. Han, X. Ge, and L. Vlacic, "Distributed Cyber Attacks Detection and Recovery Mechanism for Vehicle Platooning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 9, pp. 3821–3834, Sep. 2020, doi: 10.1109/TITS.2019.2934481.

[50] C. van der Ploeg, R. Smit, A. Siagkris-Lekkos, F. Benders, and E. Silvas, "Anomaly Detection from Cyber Threats via Infrastructure to Automated Vehicle," in *2021 European Control Conference (ECC)*, Jun. 2021, pp. 1788–1794. doi: 10.23919/ECC54610.2021.9655077.

[51] S. Xiao, X. Ge, Q.-L. Han, and Y. Zhang, "Secure Distributed Adaptive Platooning Control of Automated Vehicles Over Vehicular Ad-Hoc Networks Under Denial-of-Service Attacks," *IEEE Transactions on Cybernetics*, pp. 1–13, 2021, doi: 10.1109/TCYB.2021.3074318.

[52] W. Wei, H. Song, H. Wang, and X. Fan, "Research and Simulation of Queue Management Algorithms in Ad Hoc Networks Under DDoS Attack," *IEEE Access*, vol. 5, pp. 27810–27817, 2017, doi: 10.1109/ACCESS.2017.2681684.

[53] A. A. Alsulami, Q. Abu Al-Haija, A. Alqahtani, and R. Alsini, "Symmetrical Simulation Scheme for Anomaly Detection in Autonomous Vehicles Based on LSTM Model," *Symmetry*, vol. 14, no. 7, p. 1450, Jul. 2022, doi: 10.3390/sym14071450.

[54] M. Basnet and M. H. Ali, "A Deep Learning Perspective on Connected Automated Vehicle (CAV) Cybersecurity and Threat Intelligence," p. 21, 2021.

[55] S. Iqbal, P. Ball, M. H. Kamarudin, and A. Bradley, "Simulating Malicious Attacks on VANETs for Connected and Autonomous Vehicle Cybersecurity: A Machine Learning Dataset," p. 12, 2022.

[56] X. Chen, Z. Li, Y. Yang, L. Qi, and R. Ke, "High-Resolution Vehicle Trajectory Extraction and Denoising From Aerial Videos," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 5, pp. 3190–3202, May 2021, doi: 10.1109/TITS.2020.3003782.

[57] T. Liu, Z. Li, P. Liu, C. Xu, and D. A. Noyce, "Using empirical traffic trajectory data for crash risk evaluation under three-phase traffic theory framework," *Accident Analysis & Prevention*, vol. 157, p. 106191, Jul. 2021, doi: 10.1016/j.aap.2021.106191.

[58] S. Biswas, I. Ghosh, and S. Chandra, "Influence of signal countdown timer on efficiency and safety at signalized intersections," *Can. J. Civ. Eng.*, vol. 44, no. 4, pp. 308–318, Apr. 2017, doi: 10.1139/cjce-2016-0267.

[59] J. Henry, "Honda, Ohio aim to make smart-mobility corridor even smarter," *Automotive News*, Dec. 09, 2021. https://www.autonews.com/mobility-report/honda-ohio-aim-make-smart-mobility-corridor-even-smarter-more-connected-cars (accessed Apr. 18, 2022).

[60] K.-F. Wu, M. N. Ardiansyah, and W.-J. Ye, "An evaluation scheme for assessing the effectiveness of intersection movement assist (IMA) on improving

traffic safety," *Traffic Injury Prevention*, vol. 19, no. 2, pp. 179–183, Feb. 2018, doi: 10.1080/15389588.2017.1363891.

[61] S. Parkinson, P. Ward, K. Wilson, and J. Miller, "Cyber Threats Facing Autonomous and Connected Vehicles: Future Challenges," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 11, pp. 2898–2915, Nov. 2017, doi: 10.1109/TITS.2017.2665968.

[62] M. Islam, M. Chowdhury, H. Li, and H. Hu, "Cybersecurity Attacks in Vehicle-to-Infrastructure Applications and Their Prevention," *Transportation Research Record*, vol. 2672, no. 19, pp. 66–78, Dec. 2018, doi: 10.1177/0361198118799012.

[63] T. Mecheva and N. Kakanakov, "Cybersecurity in Intelligent Transportation Systems," *Computers*, vol. 9, no. 4, p. 83, Oct. 2020, doi: 10.3390/computers9040083.

[64] E. Adams *et al.*, "Development of DSRC device and communication system performance measures recommendations for DSRC OBE performance and security requirements.," no. FHWA-JPO-17-483, May 2016, [Online]. Available: https://rosap.ntl.bts.gov/view/dot/31627

[65] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *PROCEEDINGS OF THE IEEE*, vol. 77, no. 2, p. 30, 1989.

[66] L. R. Rabiner and B. H. Juang, "An Introduction to Hidden Markov Models," p. 12, 1986.

[67] Y. Li, F. Wang, H. Ke, L. Wang, and C. Xu, "A Driver's Physiology Sensor-Based Driving Risk Prediction Method for Lane-Changing Process Using Hidden Markov Model," *Sensors*, vol. 19, no. 12, p. 2670, Jun. 2019, doi: 10.3390/s19122670.

[68] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains," *The annals of mathematical statistics*, vol. 41, no. 1, pp. 164–171, 1970.

[69] P. A. Devijver, "Baum's forward-backward algorithm revisited," *Pattern Recognition Letters*, vol. 3, no. 6, pp. 369–373, Dec. 1985, doi: 10.1016/0167-8655(85)90023-6.

[70] K. Xie, K. Ozbay, H. Yang, and C. Li, "Mining automatically extracted vehicle trajectory data for proactive safety analytics," *Transportation Research Part C: Emerging Technologies*, vol. 106, pp. 61–72, Sep. 2019, doi: 10.1016/j.trc.2019.07.004.

[71] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data Via the EM Algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977, doi: 10.1111/j.2517-6161.1977.tb01600.x.

[72] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal Processing Magazine*, vol. 13, no. 6, pp. 47–60, Nov. 1996, doi: 10.1109/79.543975.

[73] A. Churbanov and S. Winters-Hilt, "Implementing EM and Viterbi algorithms for Hidden Markov Model in linear memory," *BMC Bioinformatics*, vol. 9, no. 1, p. 224, Dec. 2008, doi: 10.1186/1471-2105-9-224.

[74] S. Jeong, Y. Kang, J. Lee, and K. Sohn, "Variational embedding of a hidden Markov model to generate human activity sequences," *Transportation Research Part C: Emerging Technologies*, vol. 131, p. 103347, Oct. 2021, doi: 10.1016/j.trc.2021.103347.

[75] D. G. Tzikas, A. C. Likas, and N. P. Galatsanos, "The variational approximation for Bayesian inference," *IEEE Signal Processing Magazine*, vol. 25, no. 6, pp. 131–146, Nov. 2008, doi: 10.1109/MSP.2008.929620.

[76] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Networks*, vol. 18, no. 5–6, pp. 602–610, Jul. 2005, doi: 10.1016/j.neunet.2005.06.042.

[77] H. Sun, M. Chen, J. Weng, Z. Liu, and G. Geng, "Anomaly Detection for In-Vehicle Network Using CNN-LSTM With Attention Mechanism," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 10, pp. 10880–10893, Oct. 2021, doi: 10.1109/TVT.2021.3106940.

[78] H. Zhang, Z. Nan, T. Yang, Y. Liu, and N. Zheng, "A Driving Behavior Recognition Model with Bi-LSTM and Multi-Scale CNN," in *2020 IEEE Intelligent Vehicles Symposium (IV)*, Oct. 2020, pp. 284–289. doi: 10.1109/IV47402.2020.9304772.

[79] S. Sivaraman and M. M. Trivedi, "A General Active-Learning Framework for On-Road Vehicle Recognition and Tracking," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 2, pp. 267–276, Jun. 2010, doi: 10.1109/TITS.2010.2040177.

[80] S. Kumar, K. Singh, S. Kumar, O. Kaiwartya, Y. Cao, and H. Zhou, "Delimitated Anti Jammer Scheme for Internet of Vehicle: Machine Learning Based Security Approach," *IEEE Access*, vol. 7, pp. 113311–113323, 2019, doi: 10.1109/ACCESS.2019.2934632.

[81] B. He, R. Ai, Y. Yan, and X. Lang, "Accurate and robust lane detection based on Dual-View Convolutional Neutral Network," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, Jun. 2016, pp. 1041–1046. doi: 10.1109/IVS.2016.7535517.

[82] F. H. Administration, "Traffic control systems handbook: Chapter 6 detectors–FHWA office of operations.," 2018.

[83] T. Zia and U. Zahid, "Long short-term memory recurrent neural network architectures for Urdu acoustic modeling," *Int J Speech Technol*, vol. 22, no. 1, pp. 21–30, Mar. 2019, doi: 10.1007/s10772-018-09573-7.

[84] Y. Lee, H. Jeon, and K. Sohn, "Predicting Short-Term Traffic Speed Using a Deep Neural Network to Accommodate Citywide Spatio-Temporal Correlations," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 3, pp. 1435–1448, Mar. 2021, doi: 10.1109/TITS.2020.2970754.

[85] K. Lee, M. Eo, E. Jung, Y. Yoon, and W. Rhee, "Short-Term Traffic Prediction With Deep Neural Networks: A Survey," *IEEE Access*, vol. 9, pp. 54739–54756, 2021, doi: 10.1109/ACCESS.2021.3071174.

[86] T. Fernando, S. Denman, A. McFadyen, S. Sridharan, and C. Fookes, "Tree Memory Networks for modelling long-term temporal dependencies," *Neurocomputing*, vol. 304, pp. 64–81, Aug. 2018, doi: 10.1016/j.neucom.2018.03.040.

[87] Y. Qi and S. Ishak, "A Hidden Markov Model for short term prediction of traffic conditions on freeways,"

Transportation Research Part C: Emerging Technologies, vol. 43, pp. 95–111, Jun. 2014, doi: 10.1016/j.trc.2014.02.007.

[88] Q. He, X. Meng, R. Qu, and R. Xi, "Machine Learning-Based Detection for Cyber Security Attacks on Connected and Autonomous Vehicles," *Mathematics*, vol. 8, no. 8, p. 1311, Aug. 2020, doi: 10.3390/math8081311.

[89] M. Levi, Y. Allouche, and A. Kontorovich, "Advanced Analytics for Connected Car Cybersecurity," in *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, Jun. 2018, pp. 1–7. doi: 10.1109/VTCSpring.2018.8417690.

[90] S. Lefèvre, A. Carvalho, and F. Borrelli, "A Learning-Based Framework for Velocity Control in Autonomous Driving," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 1, pp. 32–42, Jan. 2016, doi: 10.1109/TASE.2015.2498192.

[91] Y. Zhang and Z. Lu, "Exploring semi-supervised variational autoencoders for biomedical relation extraction," *Methods*, vol. 166, pp. 112–119, Aug. 2019, doi: 10.1016/j.ymeth.2019.02.021.