This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3368109

Smart Mobility Digital Twin Based Automated Vehicle Navigation System: A Proof of Concept

Kui Wang, *Student Member, IEEE,* Zongdian Li, *Member, IEEE,* Kazuma Nonomura, *Student Member, IEEE,* Tao Yu, *Member, IEEE,* Kei Sakaguchi, *Senior Member, IEEE,* Omar Hashash, *Student Member, IEEE,* and Walid Saad, *Fellow, IEEE*

***Abstract*—Digital twins (DTs) have driven major advancements across various industrial domains over the past two decades. With the rapid advancements in autonomous driving and vehicle-to-everything (V2X) technologies, integrating DTs into vehicular platforms is anticipated to further revolutionize smart mobility systems. In this paper, a new smart mobility DT (SMDT) platform is proposed for the control of connected and automated vehicles (CAVs) over next-generation wireless networks. In particular, the proposed platform enables cloud services to leverage the abilities of DTs to promote the autonomous driving experience. To enhance traffic efficiency and road safety measures, a novel navigation system that exploits available DT information is designed. The SMDT platform and navigation system are implemented with state-of-the-art products, e.g., CAVs and roadside units (RSUs), and emerging technologies, e.g., cloud and cellular V2X (C-V2X). In addition, proof-of-concept (PoC) experiments are conducted to validate system performance. The performance of SMDT is evaluated from two standpoints: *(i)* the rewards of the proposed navigation system on traffic efficiency and safety and, *(ii)* the latency and reliability of the SMDT platform. Our experimental results using SUMO-based large-scale traffic simulations show that the proposed SMDT can reduce the average travel time and the blocking probability due to unexpected traffic incidents. Furthermore, the results record a peak overall latency for DT modeling and route planning services to be 155.15 ms and 810.59 ms, respectively, which validates that our proposed design aligns with the 3GPP requirements for emerging V2X use cases and fulfills the targets of the proposed design. Our demonstration video can be found at https://youtu.be/3waQwlaHQkk.**

***Index Terms*—smart mobility digital twin, navigation system, vehicle-to-everything, cloud and edge computing, implementation**

I. Introduction

A. Background

DIGITAL twins (DTs) enable a synergistic integration between the physical and cyber worlds, thereby becoming a catalyst for the ongoing digital transformation of our society [1]. In essence, the continuous evolution of DTs over the past two decades signifies their underlying potential to lead this digital transformation [2]. In particular, DTs possess remarkable abilities to establish dynamic digital models of the environment and bidirectional communications between the physical and cyber worlds. Thereby, many functionalities such as real-time monitoring, design, and optimization can be developed with DTs acting as their founding basis. The remarkable advancements have driven revolutionary applications into various sectors like manufacturing, agriculture, and transportation [3], [4].

At the forefront of these viable sectors, integrating DTs into the transportation and traffic fields constitutes a promising solution for critical challenges like traffic congestion and road accidents [5]. A recent report indicates that the global DT market reached USD 11.12 billion in 2022 and was predicted to grow at a compound annual growth rate (CAGR) of 37.5% until 2030 [6]. Notably, the automotive and transport segment has dominated the DT market in 2022, accounting for over 20% of its total revenue. This remarkable market performance highlights the crucial role of DTs as a cornerstone in the smart mobility infrastructure [7], [8].

Indeed, the convergence of smart mobility systems with DTs has ushered in new promising levels of autonomous driving breaking into connected and automated vehicles (CAVs), thereby introducing the concept of parallel intelligence [9], [10]. On the one hand, having these vehicles "connected" underpins the feasibility of DT deployment. In particular, vehicles outfitted with vehicle-to-everything (V2X) can share information with various external entities by using vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), and vehicle-to-cloud (V2C) communications [11]. In particular, V2C facilitates vehicle data transmission to the cloud, naturally enabling the establishment of DTs on the cloud plane [12]. On the other hand, the performances of onboard autonomous driving systems are hindered by multiple constraints, such as onboard sensing view and computing resources [13]. Henceforth, incorporating DT technologies or parallel intelligence is anticipated to overcome such challenges that have hindered the evolution of CAVs.

Despite these promising strides, a significant gap persists in validating these advantages upon integrating DTs in an end-to-end (E2E) real-world smart mobility. Indeed, the theoretical advantages of DTs, as demonstrated in [14]–[16], still require practical validation to ascertain their feasibility and impact. the overarching goal of this paper is to propose a novel smart mobility DT (SMDT) platform that can be used to establish a

This work was supported by the Japan National Institute of Information and Communications Technology (NICT) under JUNO Grant 22404, the U.S. National Science Foundation under JUNO3 Grant CNS-2210254, the Japan Science and Technology Agency (JST) under SPRING Grant JPMJSP2106, and by the Tokyo Tech Academy for Super Smart Society (SSS). (*Corresponding author: Zongdian Li, Kui Wang.*)

K. Wang, Z. Li, K. Nonomura, T. Yu, and K. Sakaguchi are with the Department of Electrical and Electronic Engineering, School of Engineering, Tokyo Institute of Technology (Email: {kuiw, lizd, nonomura, yutao, sakaguchi}@mobile.ee.titech.ac.jp).

O. Hashash and W. Saad are with the Bradley Department of Electrical and Computer Engineering, Virginia Tech, Arlington, VA 22203 USA (Email: {omarnh, walids}@vt.edu).

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3368109

TABLE I
FEATURE COMPARISON OF RELATED WORKS

Ref.	Key components			SAE	DT functions		Validation methods	Objectives (use cases)
	Cloud	vehicles	RSUs	level	Modeling	Service		
[12]	√	√	√	L2	√	√	Field trial&Simulation	Personalized adaptive cruise control (P-ACC).
[25]	√	√	√	L2	√		Field trial	Cooperative perception to create traffic DT.
[26]	√		√		√		Field trial	Cooperative vehicle-infrastructure system.
[29]		√		L4	√		Field trial	Validation framework for autonomous driving.
[33]	√	√				√	Simulation	Path planning considering unexpected events.
[34]	√	√	√		√	√	Simulation	Traffic light control system.
[37]	√	√		L2	√	√	Simulation	Cooperative driving at intersections.
[38]	√	√			√		Sandbox	Validation of multi-vehicle cooperation.
This work	√	√	√	L4	√	√	Field trial&Simulation	CAV Navigation system

real-world and real-time traffic DT, to enhance CAVs' ability to "see more and see further" in autonomous driving scenarios.

B. Related Works

As discussed above, given the benefits of DT technology, researchers have made some efforts to introduce the concept of DTs to vehicular technologies in recent years [17]–[20]. To build a real-time DT model of traffic environments and provide feedback services for CAVs, it is important to handle three technical challenges: sensing, communication, and computation. Sensing plays a pivotal role in environmental information capture, which is integral for the accurate representation and predictive analysis of complex vehicular and traffic systems. The work in [21] summarizes different ways for traffic information acquisition and traffic congestion monitoring, such as using stationary roadside sensors, probe-vehicle-based techniques, vehicular networks, and social networks. Among these various methods, stationary roadside sensors, typically referred to as roadside units (RSUs), can accurately and continuously monitor the traffic condition and traffic flow [22], [23]. Although the construction of RSUs is currently expensive due to the inclusion of some specialized and high-performance hardware like LiDARs, such cost can significantly decrease while the hardware becomes mass-market and widely adopted in autonomous driving and infrastructures of smart cities [24]. Hence, the use of sensor-equipped RSUs to establish a traffic DT has become popular in some related research. For example, in [25], multiple sensors deployed in the smart infrastructure are used to detect pedestrians and realize object-level central perception on the cloud. The authors in [26] develop a three-dimensional (3D) DT framework of an intelligent transportation system (ITS) based on roadside sensing methods. However, it is impractical to anticipate that the sensing range of RSUs can totally cover the entire traffic [27]. Consequently, a more realistic method is to enhance the utilization of vehicular onboard sensors, making CAVs also contribute to the modeling of DT, as discussed in [28]–[30].

When it comes to the communications aspect, the concept of V2X provides a collaborative approach where multiple smart entities can cooperatively perceive the traffic environment [31], [32]. Hence, V2X has found a multitude of DT-like or DT-related applications within the field of autonomous driving and advanced driving assistant systems (ADAS). The authors in [33] develop a software-defined network (SDN) to achieve global path planning for connected vehicles, based on vehicle-to-BS (V2BS) and vehicle-to-UAV (V2U) communications. In [34], authors establish a V2I network between road vehicles and traffic lights to build a traffic light DT and optimize traffic control. In our previous works [35], a software-defined vehicular network (SDVN) architecture for heterogeneous V2X is designed to realize cooperative perception and ensure safety in autonomous driving. Fundamentally, a DT represents a form of central cooperative perception, wherein sensory data is aggregated and fused at a central server. Thus, the establishment of a heterogeneous V2X network lays the groundwork for the extrapolation of the proposed SMDT platform.

In aligning with the discussed sensing and communication challenges, computation also emerges as a critical component in the DT frameworks, especially the collaboration of cloud and edge computing [36]. Since edge computing is well-suited for handling delay-sensitive tasks and can also offload the cloud computation, it is often employed in some related research for functions like environmental perception and object detection [25], as well as autonomous driving system tasks like motion planning and motion control [29]. On the other hand, the central cloud, as a data pool and high-level decision-maker, aggregates data from various smart entities and makes decisions, then provides feedback or services back to smart entities. For example, [37] demonstrates the use of a first-in-first-out slot reservation algorithm within a centralized server for cooperative driving at non-signalized intersections. In another instance, [38] presents a cloud control testbed for validating multi-vehicle cooperation and vehicle-road-cloud integration. Hence, in our platform, the synergetic integration of cloud and edge computing necessitates careful consideration.

C. Contributions

In Table I, we summarize some related works and compare the main features of their research, where the level of driving automation proposed by the Society of Automotive Engineers (SAE) is applied to assess the degree of autonomy in these studies. To our knowledge, there are few studies investigating the confluence of DT technology and autonomous driving. Additionally, these studies predominantly employ validation methods of simulation or field trials. Simulation-based research not only establishes DT models but can also derive a series of new services. However, those conducting field trials face challenges in demonstrating the feedback from the digital to the physical world, where the concept of DT is

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3368109

conceived as a simulation environment for modeling real-world entities with high fidelity. In our perspective, a DT transcends mere mapping and reconstruction of the physical space within cyberspace. It should also enable an automatic bidirectional data flow between the physical and digital worlds to fully unlock the potential of DTs. i.e., leveraging global information in DT to optimize and control physical entities.

To validate the feasibility of applying the SMDT platform within the domain of autonomous driving and explore its potential, the most important task is the system design and implementation in the real world, which poses numerous challenges. Firstly, it is essential to create a real-time virtual representation of real-world traffic environments (i.e., traffic DT), which involves sensing, perceiving, and visualizing various static and dynamic entities. Secondly, the system design should consider safety and robustness issues, because of massive data collection from traffic environments and the requirement for autonomous vehicles to operate in unpredictable conditions, which necessitates an effective solution for dynamic traffic information acquisition and a reliable V2X communication network. Thirdly, successful implementation requires a holistic consideration and integration of hardware, software, and communication, as well as collaboration between cloud and edge computing. Finally, we also need to design effective methods for functionality verification and performance assessment, which involves substantiating functional capabilities through proof-of-concept (PoC) experiments and appraising the benefits brought by SMDT platform via large-scale traffic simulations.

Hence, based on the discussions above, the main contribution of this paper is a novel SMDT platform that can provide cloud services for CAVs. In particular, we make the following key contributions:

- *Architecture design and use case*: we develop a novel system architecture of the SMDT platform designed to address safety and robustness considerations. This architecture integrates cloud and edge computing across RSUs, CAVs, and a central cloud. Then we introduce a CAV navigation system as the use case of the SMDT platform;
- *Comprehensive system design and implementation*: we detail the intricacies of core software and hardware components, as well as pivotal technologies for realizing SMDT platform in real-world autonomous driving environment. Then we achieve the implementation in the test field, including a cloud server and distributed intelligent RSU and CAV edges with sensor, communication module, and edge computing capability;
- *PoC demonstration*: we conduct a field trial on SMDT-empowered route planning service to validate the system functionalities. We also evaluate the communication reliability and latency based on standards for SSMS and information sharing V2X use cases proposed by the 3rd generation partnership project (3GPP);
- *Large-scale traffic simulation*: we develop a traffic simulation solution so as to study the benefits of the proposed SMDT platform in terms of traffic efficiency and safety when using the proposed CAV navigation system.

The rest of this paper is organized as follows. Section II explains the system architecture of SMDT platform and introduces the CAV navigation system as a use case, then we analyze the communication and latency requirements for the proposed SMDT platform and CAV navigation system. Then, the deployment intricacies, including software installation, hardware deployment, and communication establishment, are presented in Section III. Section IV demonstrates the case study in the PoC experiment to validate the system functionality. In Section V, the results from large-scale traffic simulation are discussed, and some metrics are measured to validate the system efficacy from the perspective of reliability and latency. Finally, we draw conclusions in Section VI.

II. SMDT Platform for Autonomous Driving

In this section, we introduce the architecture of the SMDT platform, in which a *traffic DT* replicates the real-time real-world traffic information in cyberspace and enables enhanced cloud services for traffic efficiency and road safety. Relying on the potential of the SMDT platform, we propose the DT-empowered CAV navigation system, of which the functionality and latency requirements are discussed in detail.

A. SMDT Platform Architecture

The proposed SMDT platform integrates the cloud and edge computing resources of ITS. This architecture is chosen to exploit the robust computational capabilities of the cloud for data-intensive tasks while utilizing edge computing to minimize response times for real-time services. This dual-computing approach is essential to manage latency, safety, and processing efficiency in large-scale traffic network operations effectively. Computing at the cloud can utilize plentiful and stronger processing units, but it introduces considerable network latency. The SMDT platform prioritizes cloud computing when handling computation-intensive tasks that require global big data from large-scale traffic networks. Computing at the edge cannot enjoy the same processing efficiency as cloud computing due to the limited power supply and space for deploying units. However, it guarantees real-time responses to the users, addressing their demands for delay-sensitive services on the SMDT platform. As depicted in Fig. 1, the computing units of the SMDT platform are categorized into three types: RSU edges, CAV edges, and a central cloud. They are interconnected and collaboratively support the platform operations, including *traffic DT* creation, data processing, and user interaction.

1) RSU Edges: RSUs play an important role in the SMDT platform, primarily focused on high-traffic areas. RSUs are equipped with a variety of sensors to monitor traffic flow and detect accidents, and they also have the capability to receive data from connected vehicles (CVs) through V2I communication, which allows for a more efficient and targeted deployment of RSUs, ensuring they are strategically placed where their impact on traffic management and safety is maximized. These two roles of RSU enhance the effectiveness of the *traffic DT* and optimize the utilization of RSUs in the overall network.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3368109

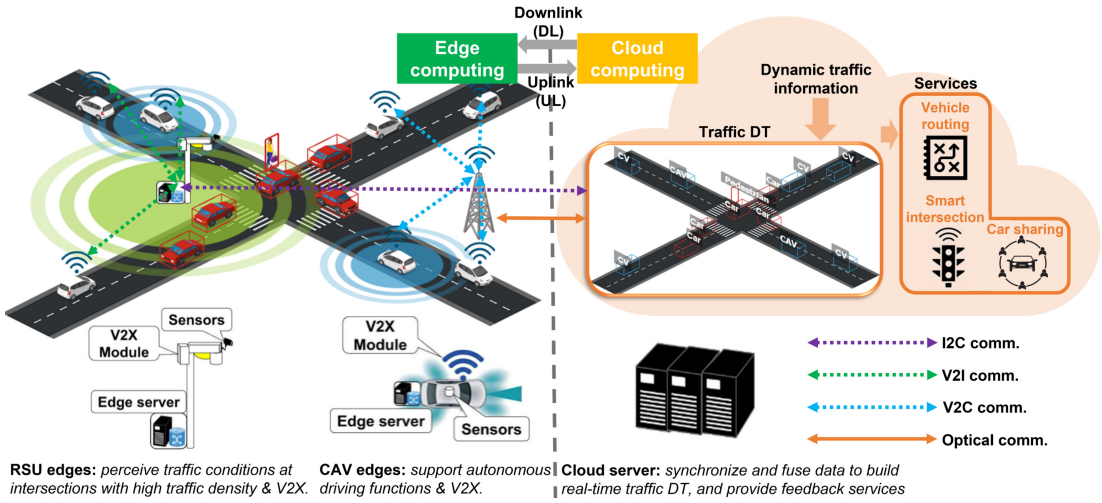


Fig. 1. SMDT high-level conceptual system architecture with cloud/edge computing.

2) CAV edges: All CAVs operate in a highly automated mode and strictly adhere to directives issued by the cloud. Besides contextual awareness, CAV sensors are used for localization, local path planning, and motion control. As a basic requirement of the SMDT platform, CAVs should report their positions to get services and upload their sensing data to compensate for the undetectable areas of RSU sensors when constructing the *traffic DT*. In essence, CAVs are the mobile counterparts of RSUs. Importantly, these CAVs should be designed with a monitoring system that can be aware of the signal loss and switch to standalone mode for decision-making, ensuring that even if a CAV temporarily loses its communication link, it can still maintain safety and operational integrity until the connection is re-established.

3) Central cloud: The cloud is the place where a dynamic *traffic DT* is established by integrating a comprehensive array of traffic data. This integration encompasses real-time information from both RSU and CAV edges, along with inputs from various commercial traffic information providers. Serving as a vital resource pool, the cloud processes and analyzes this diverse data to offer a detailed and expansive view of traffic conditions. Such a robust integration of multiple data sources is instrumental for both providing accurate traffic analysis and ensuring the robustness of the system, thereby enhancing management solutions for road safety and traffic efficiency.

Data sharing between cloud and edge computing can be divided into downlink (DL) and uplink (UL). The edge servers at the RSU and CAV continuously upload various levels of real-time sensing data, such as raw data or processed data, according to different use cases. The cloud provides services and issues directives over the DL to edges. To ensure connectivity among these smart entities and reliable DL/UL between cloud and edge, it is necessary to establish a V2X network, including V2C, V2I, and infrastructure-to-cloud (I2C) communications, where V2C communication is designed with a multi-radio access technology (Multi-RAT) handover mechanism based on channel conditions to ensures consistent and stable communication. V2C communication ensures the reach to the cloud through cellular networks or relayed by RSUs using dedicated short-range communications (DSRC), millimeter-wave (mmWave), and Wi-Fi.

B. Use Case: CAV Navigation System

To evaluate and highlight the benefits of the proposed SMDT platform from the practical viewpoint of applications and services, a CAV navigation system is designed based on SMDT platform. In a dynamic traffic environment, road segments may experience unexpected incidents—such as accidents, large gatherings, or peak-time congestion—that can drastically affect journey time and road safety. Therefore, it is crucial to devise a mechanism that continuously detects such incidents and dynamically re-routes users.

Hence, the CAV navigation system is designed to employ an event-triggered planning strategy. As users enter the road network, the optimal routes are generated based on current traffic conditions to minimize journey time from origin to destination. During transit, if specific events occur within the traffic network, such as a car crash, a route re-planning service is triggered to help users avoid these events.

1) Journey Time Calculation: We use a directed graph $G = (N, L)$ to represent the traffic network. N is the set of nodes (i.e., traffic intersection) and $N = \{n^i\}_{i=1}^{M}$, where M is the total number of nodes. L is defined as the set of directed links (i.e., unidirectional traffic roads) and $L = \{l^{i,j} = (n^i \rightarrow n^j), n^i, n^j \in N\}$, where $l^{i,j}$ means the directed road from n^i to n^j. As we can obtain the real-time traffic volume x on the road link l with the proposed SMDT platform, based on the traffic volume and road length s, we can calculate the traffic density k with:

$$k = \frac{x}{s} \tag{1}$$

According to [39], [40], the relationship between traffic density k and journey speed v can be modeled as

$$v = v_{\text{free}}(1 - \frac{k}{k_{\text{max}}}) \tag{2}$$

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3368109

where v_f is the free-flow speed, and $k_{\max}$ is the maximum vehicle density. Thus, the journey time can be calculated as:

$$t_{\text{jry}} = \frac{s}{v} = \frac{s}{v_{\text{free}}(1 - \frac{k}{k_{\max}})} \tag{3}$$

Based on equation (3), we use a matrix T_{jry} to represent the node-to-node journey time for each road segment within the traffic network:

$$\boldsymbol{T}_{\text{jry}} = \begin{array}{c} \\ n^1 \\ n^2 \\ \vdots \\ n^M \end{array} \begin{array}{c} \begin{array}{cccc} n^1 & n^2 & \cdots & n^M \end{array} \\ \begin{pmatrix} t^{1,1}(t) & t^{1,2}(t) & \cdots & t^{M,M}(t) \\ t^{2,1}(t) & t^{2,2}(t) & \cdots & t^{M,M}(t) \\ \vdots & \vdots & \ddots & \vdots \\ t^{M,1}(t) & t^{M,2}(t) & \cdots & t^{M,M}(t) \end{pmatrix} \end{array} \tag{4}$$

where M is the total number of network nodes. If there exists no link from node n^i to node n^j, then the journey time $t^{i,j}$ will be set to $+\infty$.

2) Event-triggered Mechanism: Traffic event is defined as unexpected and sporadic scenarios that have severe impacts on traffic efficiency and safety. In this study, we primarily analyze the two typical scenarios along with their detection criteria:

- *Pedestrian gathering:* To detect the occurrence of overcrowded gatherings, we introduce a pedestrian density threshold d_{thre} as the criteria for assessment. When the measured density d^i in the vicinity of the intersection n^i is higher than d_{thre}, the intersection n^i will be regarded as overcrowded.
- *Traffic accidents:* Vehicular speeds can be regarded as a criterion of the accident occurrence. Specifically, if a particular link $l^{i,j}$ or intersection n^i exhibits vehicles with speeds consistently below a threshold v_{thre}, it is inferred that the link $l^{i,j}$ or the intersection n^i has become a locus of the traffic accident.

In the traffic DT-based traffic monitoring process, we continuously update the sets of nodes N_{eve} and links L_{eve} that are influenced by the aforementioned two kinds of events.

3) Workflow of CAV Navigation System: Based on the computation of *journey time* and the *event-triggered mechanism*, the workflow of CAV navigation system can be designed as a cooperative event-triggered route planning process. The cooperative aspect is manifested by the necessity for all CAV users to upload their sensor data. This collaborative data sharing serves to effectively compensate for areas not covered by RSU sensors. On the other hand, the *event-triggered mechanism* comes into play when traffic events are detected within the network. Such incidents then trigger route re-planning for users whose original routes overlap with these events.

The overall operation of the proposed route planning algorithm is shown in Algorithm 1. The procedure operates on the road network G and the predicted journey $T_{\textbf{jry}}$. For the CAV users just entering the network, denoted as U_{new}, their initial route is generated using Dijkstra algorithm. Here, we note that the Dijkstra algorithm is employed to generate the fastest path, which is facilitated by assigning each road link's journey time as its weight. If some events are detected at intersections, the journey time for links pointing towards these intersections will be set to infinity, to help users circumvent such events. Similarly, if events happen on specific road links, their journey time will also be set to infinity. Then the planned routes of CAV users should be re-evaluated. If any part of a user's planned route includes a link with an infinite journey time, the user's route is recalculated using Dijkstra algorithm, which ensures that all users are always provided with an optimal route considering the latest information.

Algorithm 1 Cooperative Event-triggered Route Planning

Input:
$G = (N, L)$ // Road network
$\boldsymbol{T}_{\text{jry}}$ // Journey time of each road link
$U = \{u^i\}_{i=1}^{M_U}$ // Set of CAV user already in road network
$U_{\text{new}} = \{u_n^i\}_{i=1}^{M_N}$ // Set of new CAV user entering road network
$R(u_i)$ // Planned route of each user already in road network
N_{eve} // Set of intersections with detected event
L_{eve} // Set of roads with detected event
Output:
$R_{\text{new}}(u_i)$ // Re-planned route for each user
Initialization: $R_{\text{new}} \leftarrow \emptyset$ // Plan route for new entering CAVs.
for $u_n^i \in U_{new}$ **do**
  $n_{\text{start}} := u_n^i.CurrentPosition$
  $n_{\text{end}} := u_n^i.Destination$
  $R_{\text{new}}(u_n^i) :=$ Dijkstra$(N, L, \boldsymbol{T}_{\text{jry}}, n_{\text{start}}, n_{\text{end}})$
end
// Set the journey time of links pointing to the nodes, where the events occurred, to infinity.
for $n^i \in N_{eve}$ **do**
  for $n^j \in N_{eve} \setminus \{n^i\}$ **do**
    if $\boldsymbol{T}_{jry}(n^j, n^i) \neq +\infty$ **then**
      $\boldsymbol{T}_{\text{jry}}(n^j, n^i) := +\infty$
    end
  end
end
// Set the journey time of links with events to infinity.
for $l^{i,j} \in L_{eve}$ **do**
  $\boldsymbol{T}_{\text{jry}}(n^i, n^j) := +\infty$
end
// Re-plan route for users.
for $u^i \in U$ **do**
  for $l^{i,j} \in R(u_i)$ **do**
    if *there exists* $\boldsymbol{T}_{jry}(n^j, n^i) = +\infty$ **then**
      $n_{\text{start}} := u^i.CurrentPosition$
      $n_{\text{end}} := u^i.Destination$
      $R_{\text{new}}(u^i) :=$ Dijkstra$(N, L, \boldsymbol{T}_{\text{jry}}, n_{\text{start}}, n_{\text{end}})$
    end
  end
end

C. Requirements for SMDT-based CAV Navigation Use Case

The SMDT-based CAV navigation system features a centralized architecture focusing on two core processes: *traffic DT modeling* and *route planning service*. To achieve *traffic DT modeling* over the cloud plane with RSU and CAV sensing data, sensor-equipped RSUs are adopted for collecting raw data nearby traffic intersections, and CAVs with onboard

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3368109

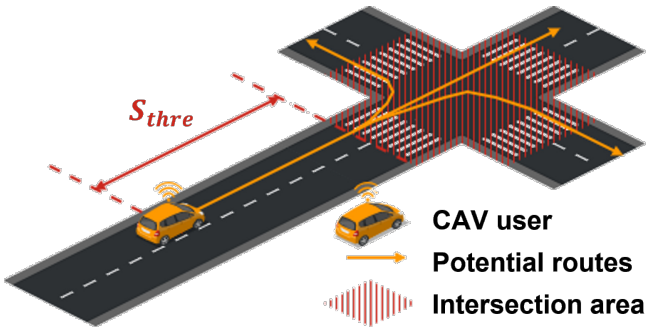


Fig. 2. Illustration of route-planning request distance.

TABLE II
V2X USE CASES AND RESPECTIVE KPIS

Use cases	Reliability (%)	E2E latency (ms)	Total latency
SSMS	95	10	$T_{\text{dt}} \ll T_{\text{svc}}$
Information Sharing	-	100	$T_{\text{svc}} < 0.164\ v_{\text{free}}$

sensors driving within the road network serve to supplement traffic conditions beyond the sensing range of RSUs. As for *route planning service*, it is important to establish reliable communication to ensure the dissemination of planned routes. Upon near entering into the road network, the user should "tell" the cloud its impending presence, initiating continuous uploads of its position and desired destination, and then the cloud responds by furnishing an initially planned route. Since the initial planned route might make the user alter their predetermined direction at an intersection, as shown in Fig. 2, the request should be initiated prior to the user entering the intersection. Here, we define a threshold distance S_{thre} at intersections, i.e., when the distance between the CAV user and the intersection area reaches S_{thre}, the user will send a route planning request to the cloud.

The threshold distance S_{thre} is determined with the consideration of comfort issues. The vehicle should have enough time to process the planned route before reaching the intersection. Given the presence of pedestrians crossing at intersections, it is necessary for the vehicle to comfortably decelerate and stop before reaching the intersection. According to the Institute of Transportation Engineers (ITE) recommendations, a comfortable deceleration rate a_{comfy} should be less than 10 ft/s^2, equivalent to 3.048 m/s^2 [41]. Hence, S_{thre} can be decided with the maximum speed (i.e., free-flow speed v_{free}) as:

$$S_{\text{thre}} = \frac{v_{\text{free}}^2}{2a_{\text{comfy}}} = 0.164\ v_{\text{free}}^2 \tag{5}$$

Communication links in the *traffic DT modeling* and *route planning service* processes correspond to two V2X use cases proposed by 3GPP [42], i.e., SSMS and information sharing for high/full automated diving, respectively. SSMS enables the sharing of raw or processed sensor data to build collective situational awareness, which allows the central server to access and gather real-time traffic information obtained by RSU sensors. The information sharing use case encompasses two scenarios: cooperative perception and cooperative maneuver. Cooperative maneuver necessitates the sharing of detailed planned trajectories among all participating vehicles via V2X communication, which permits the central server to exert control over the vehicular routes.

To assess whether the proposed SMDT platform can meet the necessary requirements of the above two use cases, the respective KPIs are summarized in Table II. The requirements in terms of reliability and E2E latency are defined in the same way as 3GPP technical report (TR) 22.886 [43]. By definition, reliability refers to the probability of successful transmission reliability and E2E latency refers to the one-way time delay it takes to deliver a piece of information from a source to a destination, without the application-layer processing delay. In this study, E2E latency of SSMS and information sharing use cases are defined respectively as the unidirectional communication delay between RSUs and the cloud, termed as T_{I2C}, and between the CAVs and the cloud, termed as T_{V2C}. In addition, we also define the total latency of *traffic DT modeling* and *route planning* processes. In our system, the basis for decision-making is the dynamic traffic DT, which must represent real-time traffic conditions. Consequently, the latency in DT modeling T_{dt} needs to be significantly shorter than the latency in the route planning service process T_{svc}, to ensure that the traffic DT used for decision-making can be regarded as real-time. The DT modeling latency T_{dt} mainly consists of RSU edge computational time and one-way E2E latency from LiDAR to cloud server, expressed as:

$$T_{\text{dt}} = T_{\text{RSU}} + T_{\text{I2C}} \tag{6}$$

Considering the route planning service is based on a request-response mechanism, it is imperative to ensure that users can receive, load, and execute the new planned route before entering the intersection. The route planning latency T_{svc}, spanning from the moment the user reaches the request distance S_{thre} to the execution of the route, should meet:

$$T_{\text{svc}} \leq \frac{S_{\text{thre}}}{v_{\text{free}}} = 0.164\ v_{\text{free}} \tag{7}$$

where the latency of offering route planning service T_{svc} includes time delays caused by localization T_{local}, route loading and execution T_{exe} in the autonomous driving system, cloud computing T_{cloud}, and bidirectional V2C communication T_{V2C}.

$$T_{\text{svc}} = T_{\text{local}} + T_{\text{exe}} + T_{\text{cloud}} + 2\ T_{\text{V2C}} \tag{8}$$

III. DESIGN AND IMPLEMENTATION OF SMDT PLATFORM

In this section, we design an example system architecture of the proposed SMDT platform tailored for implementation, as shown in Fig. 3. In the vein of conceptual design, the example design also includes three key components: RSU, CAV, and the central cloud. The primary objective of this implementation-tailored system is to realize the conception of SMDT platform depicted in Fig. 1, utilizing LiDAR sensing, cloud-edge computing, and V2X communication, which ensures that the *traffic DT modeling* and *route planning service* can be developed and deployed in the real-world traffic environment.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3368109

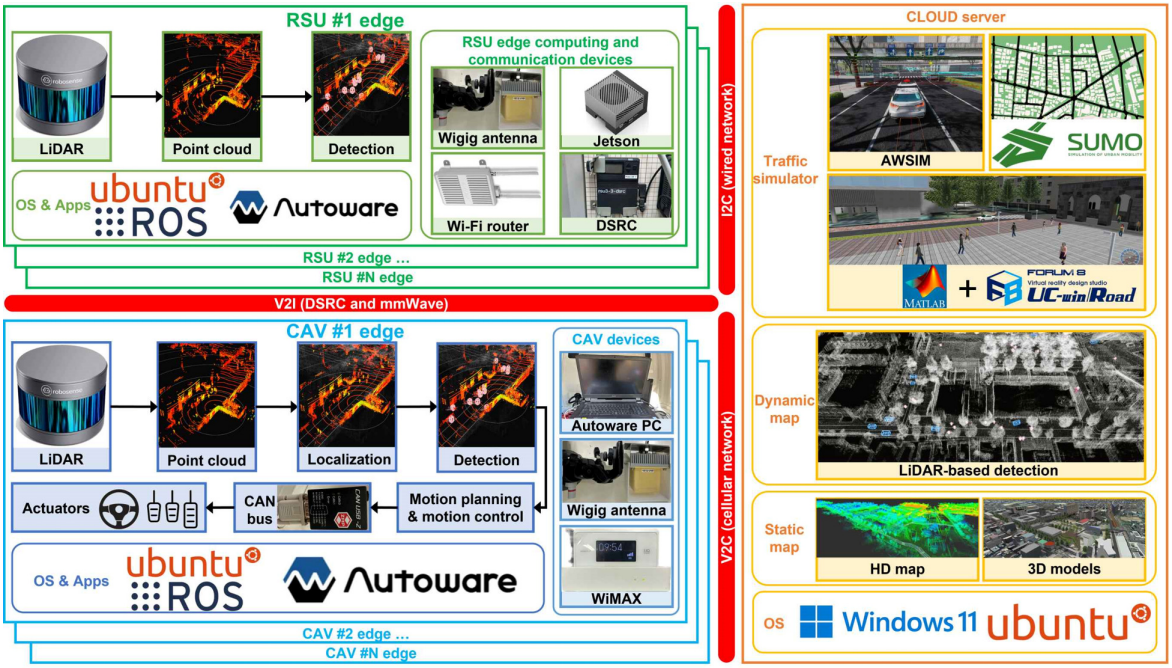


Fig. 3. Example system design of the SMDT platform.

TABLE III
COMMUNICATION REQUIREMENTS AND PERFORMANCES

Function	Requirements	Message contents	Methods	Performances
V2C	Data rate: $\geq$ 80 Mbps Coverage: $\geq$ 1 km	Uplink: vehicle position and processed sensor data. Downlink: cloud-based decisions	WiMAX	$\geq$ 120 Mbps $\geq$ 50 km
V2I	Data rate: $\geq$ 1 Mbps Coverage: $\geq$ 50 m	Cooperative perception (processed data, e.g., detection and tracking result)	Wi-Fi router	$\geq$ 10 Mbps $\geq$ 200 m
	Data rate: $\geq$ 1 Gbps Coverage: $\geq$ 20 m	Cooperative perception (raw sensor data, e.g. LiDAR point cloud and camera images)	WiGig	$\geq$ 1 Gbps $\geq$ 120 m
I2C	Data rate: $\geq$ 1 Gbps	Environmental perception (both raw data and processed data)	Ethernet	$\geq$ 1 Gbps

TABLE IV
HARDWARE DEVICES AND SPECIFICATIONS

Type	Description		
	Device names	Specifications	Amount
CAV	RoboCar (CAV #1)	Automated vehicle with driving controller	1
	MiniVan (CAV #2)	Automated vehicle with driving controller	1
Sensor	RS-LiDAR-32	Location: CAV edges, Range: 200 m, Accuracy: $\pm$ 3 cm, Rotation Speed:10/20 Hz	2
	RS-LiDAR-80	Location: RSU edges, Range: 200 m, Accuracy: $\pm$ 3 cm, Rotation speed: 5/10/20 Hz	3
Cloud/ edge servers	Autoware PC	Location: CAV edge (RoboCar), OS: Ubuntu 16.04, ROS: Kinetic, Autoware. AI	1
	Autoware PC	Location: CAV edge (MiniVan), OS: Ubuntu 22.04, ROS: Foxy, Autoware. Universe	1
	Jetson AGX Orin	Location: RSU edges, OS: Ubuntu 20.04 (JetPack 5.0.2), ROS: Galactic, Autoware. Universe	3
	DT engine	Location: cloud server, OS: Ubuntu 20.04, ROS: Galactic, Autoware. Universe	1
	DT simulator	Location: traffic simulator, OS: Windows 11, Simulation app: SUMO, UC-win/Road	1

A. Hardware Deployment and Heterogeneous V2X Network

Our SMDT testing environment is named Tokyo Tech Smart Mobility R&E Field[1], as shown in Fig. 4. In the field setting, there are three RSUs installed at three corners of a square road section, two CAVs that can achieve high autonomy, as well as the remote cloud server located far away from the testing field. The communication system within the SMDT platform is based on a heterogeneous V2X network and managed by an SDN controller [35], which enables three kinds of V2X links: V2I, V2C, and I2C communications. Considering the large communication distance between vehicles and the cloud, the cellular network is applied for V2C communication. Depending on the application scenario, RSUs may send data of different levels to the cloud server and nearby vehicles, such as raw data or processed data, which have distinct requirements for communication coverage and bandwidth. Hence, we use wired networks for I2C communication and employ both DSRC and mmWave technologies for V2I. Hence, each RSU and CAV are equipped with communication modules, including WiMAX for V2C communication, a Wi-Fi router as a replacement for DSRC, WiGig antenna for mmWave communication, and

[1]https://www.wise-sss.titech.ac.jp/en/.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3368109

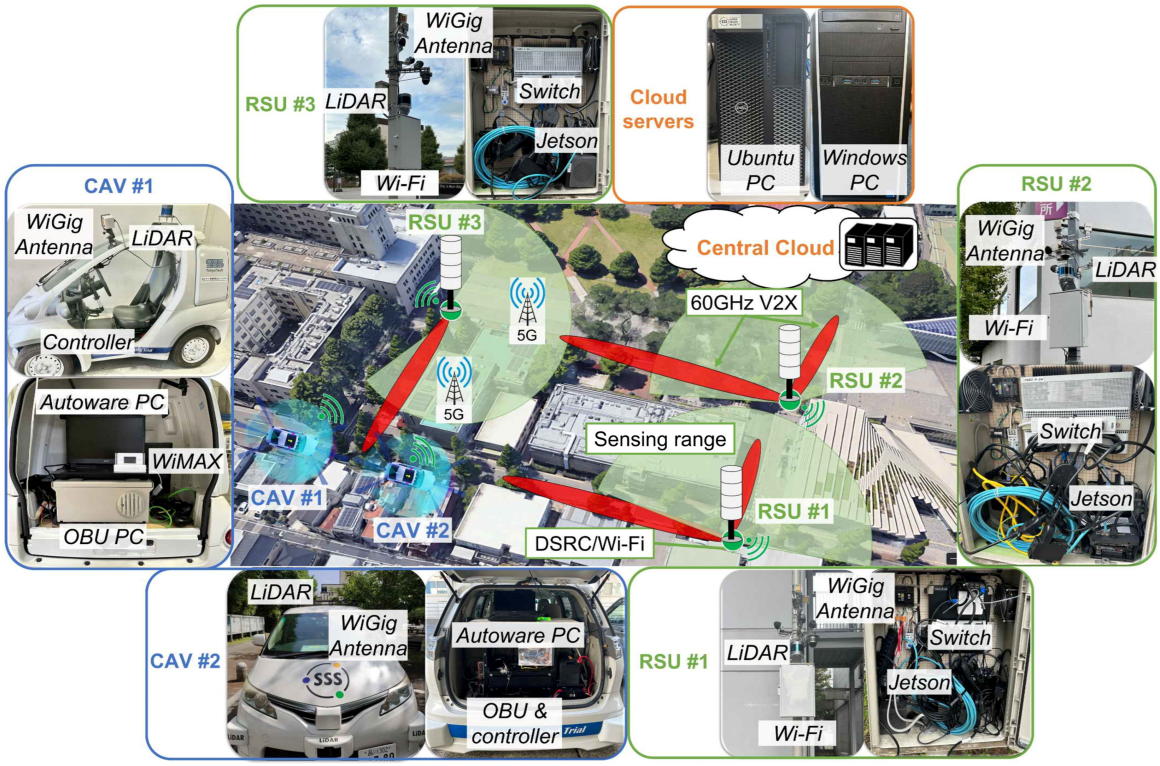


Fig. 4. Hardware components in smart mobility R&E field of Tokyo Tech Academy for Super Smart Society.

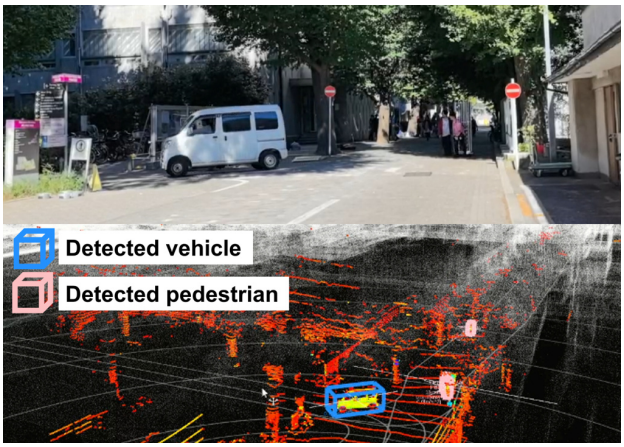


Fig. 5. An example of LiDAR-based object detection in the world.

Ethernet cables for wired communication through the campus network. In Table III, we list the required communication speeds and coverage in our experiment field, as well as the performance of selected communication methods.

As for sensing and computing devices, summarized in Table IV, an 80-layer LiDAR is integrated alongside an NVIDIA Jetson in each RSU. LiDAR is responsible for capturing raw data (i.e., dynamic point clouds) from the physical world. The Jetson, on the other hand, is employed to implement specific functional modules, such as object detection and tracking based on point clouds, thereby enabling object-level projection from the physical world to the digital world. Two CAVs are outfitted with 32-layer LiDAR sensors positioned on the rooftop to sense their surroundings. A dedicated Autoware PC is utilized for processing the LiDAR data, facilitating environmental perception, localization, motion planning, and motion control. The control signals are then transmitted to the onboard unit (OBU) to achieve autonomous driving. The remote cloud server works as the central hub for data aggregation and storage, global information processing, and providing feedback services to the autonomous vehicle. Therefore, the performance requirements for the cloud server are very high, including scalability, processing power, and reliability to efficiently and securely handle large volumes of real-time data. Consequently, we select a computer equipped with a GeForce RTX 5000 GPU and ensure abundant memory and storage resources to satisfy the computational demands.

B. Software Installation and Traffic DT Modeling

The software in our platform is centered around Autoware [44] and Robot Operating System (ROS). Autoware facilitates the detection and tracking of traffic participants within the RSU sensor range. The detection module applies the Center-Point framework [45], which can detect, identify, and visualize 3D objects from the LiDAR point clouds in real time. Then, a Multi-object Tracker module [46], is responsible for assigning the detected objects with IDs and estimating their velocities. An example of detection and tracking results is shown in Fig. 5. Additionally, a Normal Distributions Transform (NDT) algorithm is utilized for the LiDAR scan matching with the 3D point cloud map, which enables real-time localization with centimeter-level accuracy. The driving route is represented in the form of Waypoints, which includes a set of points

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3368109

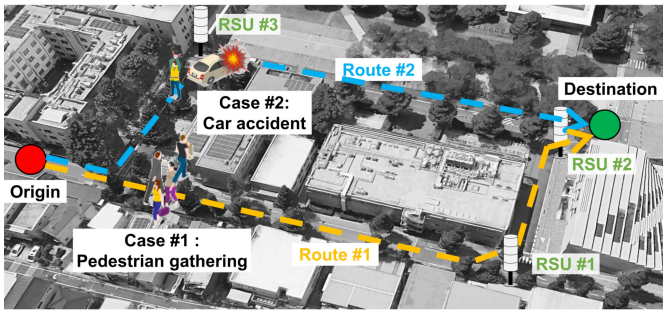


Fig. 6. An overview of experiment setup.

with coordinates and desired speed. To perform local motion planning and motion control, a velocity planner adjusts the velocity based on the Waypoints to decelerate or accelerate in response to nearby objects and road characteristics, such as stop lines and traffic lights. Finally, a Pure Pursuit algorithm is employed to generate coordinated velocities and steering angles and follow the target Waypoints.

Over the cloud plane, since ROS facilitates seamless communication among distributed computers within a Local Area Network (LAN), the cloud server can easily access all ROS2-defined messages from RSU edges. Besides, the Hypertext Transfer Protocol (HTTP) communication is utilized to establish the V2C link, enabling the real-time detection results (a kind of ROS2-defined massages) uploading and planned route downloading on CAV edges. After acquiring detection results from RSUs and CAVs, we can fuse them and build the real-time object-level *traffic DT* in Autoware visualization tool Rviz. Then, the cloud server utilizes objects' positions to align them with the corresponding road segments, which facilitates traffic monitoring on each road section, thereby realizing *journey time calculation* and *event-triggered mchanism* mentioned in Section II.B. Additionally, for large-scale traffic experiments that cannot be executed within our testing field, we also create the simulation environment with some traffic simulation software, such as SUMO [47] and UC-win/Road [48].

IV. Proof-of-Concept Demonstration of CAV Navigation System

In this section, we present an outdoor PoC experiment of SMDT-based route planning for autonomous driving. The experimental setup for route planning process in the Tokyo Tech Smart Mobility Field is first explained. Then we study two practical cases that effectively reflect key functions of proposed system.

A. PoC Setup in Tokyo Tech Smart Mobility Field

Given the context of our Smart Mobility Field located within a campus environment, it inherently experiences minimal vehicular flow. This means that under normal conditions, vehicle volume is not a determinant of journey times. However, the driving environment is relatively complex due to the frequent presence of pedestrians and cyclists on the roadways. Coupled with a simple road network structure, our PoC experiment does not emphasize the *journey time calculation* function or the usage of the Dijkstra algorithm to search for a route for autonomous vehicles. Instead, the focus is on demonstrating the *event-triggered mechanism*, i.e., utilizing the established *traffic DT* to detect and identify traffic events, then subsequently providing route planning for the CAV to bypass such incidents.

TABLE V
Roles of CAVs, RSUs, and the Cloud in PoC Experiment.

Component	Functionalities
RSU	1. Detect and track objects within sensing range; 2. Detect traffic accidents within sensing range.
CAV	1. Operate in a fully autonomous driving mode; 2. Detect and track objects within sensing range; 3. Detect traffic accidents within sensing range.
Cloud server	1. Collect traffic information from RSUs and CAVs; 2. Model and visualize DT; 3. Assign routes for CAVs to avoid accidents.

In this experiment, the ego vehicle corresponds to CAV #1 as shown in Fig. 4. Its origin and destination are depicted in Fig. 6. There are two routes with approximately equal distances bridging the start and end points: Route #1 (206 m) and Route #2 (210 m), where Route #1 is regarded as a default route. Under normal conditions without any traffic events, the cloud server would select Route #1 and send its Waypoint file to CAV users by default. In Case #1, A *pedestrian gathering* event occurs on the default Route #1, which is not within the sensing range of any RSU. Both CAV #1 (i.e., ego vehicle) and CAV #2 drive through the road network sequentially from origin to destination, where the maximum speed of CAVs $v_{\max}$ are set to match the road speed limit, i.e., 20 km/s in the campus. As CAV #2 first enters the network and travels along the default Route #1, it will encounter the *pedestrian gathering* and stop in front of it at a safe distance. This event will be thus detected by CAV #2 sensor. In Case #2, we park CAV #2 on Route #2 to simulate the occurrence of a *traffic accident*, positioned within the sensing range of RSU #3. Then this event can be detected by RSU #3. Some main responsibilities of RSUs, CAVs, and the cloud server in the PoC experiment are summarized in Table V.

B. Case Study: Traffic DT-based Route Planning

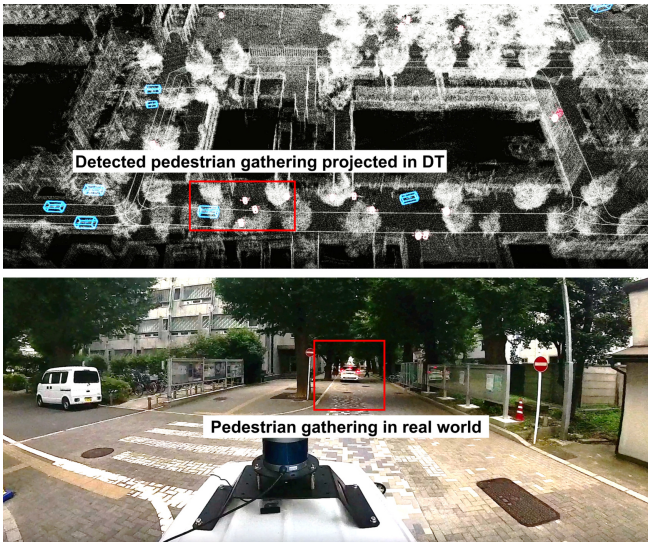
According to these two cases, a PoC experiment is conducted and a demonstration video[2] is available online, showcasing the practical implementation and effectiveness of the SMDT-based CAV navigation system in real-world scenarios.

1) Case #1 Pedestrian gathering detected by CAV: Fig. 7 shows the results of SMDT-based route planning process in Case #1, Fig. 7(a) gives a view of traffic DT on cloud server, where the pink bounding boxes represent detected pedestrians. It can be seen that there are three pedestrians walking on the roadway of Route #1. This information is then captured by CAV #2, which is the first CAV entering the road network, and subsequently uploaded to the cloud. A car-following camera mounted on CAV #1 to record its driving process, also captures

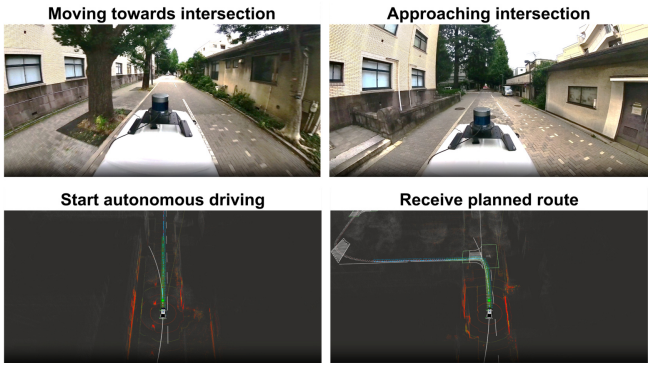
[2]https://youtu.be/3waQwlaHQkk

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3368109

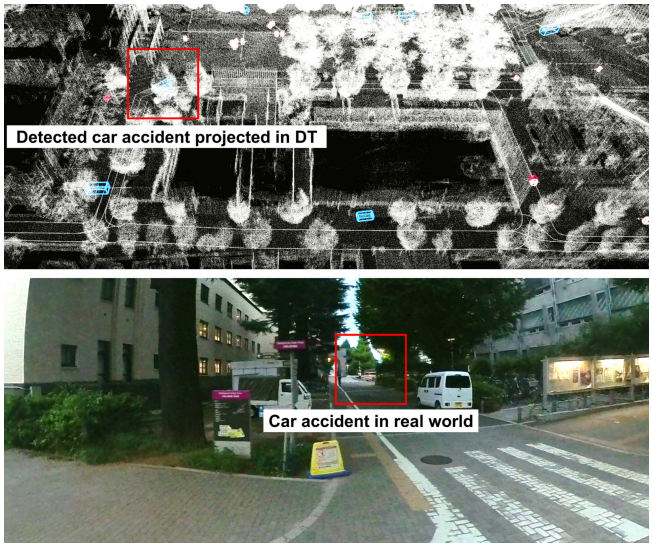


(a)

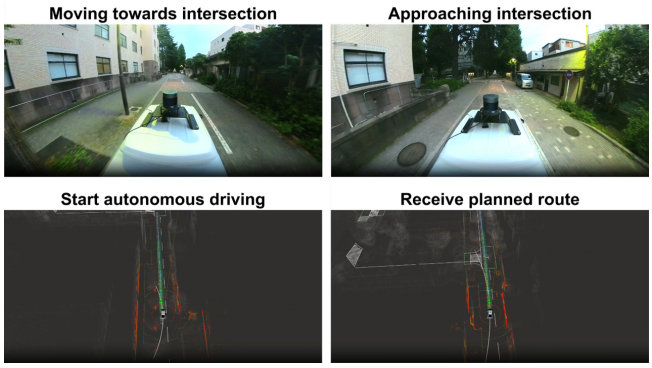


(b)

Fig. 7. Illustration of traffic DT and autonomous driving operation results in Case #1. (a): visualization of traffic DT and real-world image showing the congestion caused by pedestrian gathering, (b) car-following camera view and ego vehicle Rviz view showing the executed route.



(a)



(b)

Fig. 8. Illustration of traffic DT and autonomous driving operation results in Case #2. (a): visualization of traffic DT and real-world image showing the congestion caused by pedestrian gathering, (b) car-following camera view and ego vehicle Rviz view showing the executed route.

this scene, where CAV #2 halts in front of the gathering crowd. Fig. 7(b) shows the autonomous driving process of CAV #1 from the perspectives of the car-following camera and the Autoware visualization tool Rviz, respectively. During the initialization phase, CAV #1 is proceeding straight, poised to enter the road network. When CAV #1 arrives at the predetermined threshold distance S_{thre} from the intersection area, it will start sending its position to the cloud through V2C communication to notify its impending entry into the network and requests route planning service. Then the cloud sends the Route #2 's Waypoints file to CAV #1 to help it avoid pedestrian gathering area and navigate it to destination. Consequently, CAV #1 executes the planned route to turn left at the intersection and finally arrives at the destination. In this experiment, the threshold distance S_{thre} is determined according to equation (5) with maximum vehicle speed:

$$S_{\text{thre}} = 0.164\ v_{\text{max}}^2 = 5.062\ m \tag{9}$$

2) Case #2 Traffic accident detected by RSU: Similar to Case #1, Fig. 8(a) shows the traffic DT, where a parked car can be found on Route #2, represented by a blue bounding box. Under such circumstances, the cloud system would deduce that a car accident has occurred on Route #2. In Fig. 7(b), we also give a real-world image from car-following camera on CAV #1 that records the parking car and executed route in Rviz. In this case, the cloud sends Route #1 's Waypoints to CAV #1 to help it avoid the car accident on Route #2.

The case study demonstrates the effectiveness of SMDT-based route planning for autonomous vehicles: *(i)* the system's ability to detect and respond to real-time events, such as pedestrian gatherings and traffic accidents, and *(ii)* reliable route planning service to achieve the control on vehicle level. This underscores the system's potential for real-world application in dynamic traffic environments. However, challenges for future development mainly include safety and robustness issues. Safety concerns primarily arise from the reliance on the V2X communication network and the accuracy of traffic monitoring, especially in unpredictable urban environments with high CAV density. Robustness is challenged by varying environmental conditions and sensor limitations. To address these, future work could: *(i)* enhance V2X network resilience by implementing advanced encryption and cybersecurity pro-

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3368109

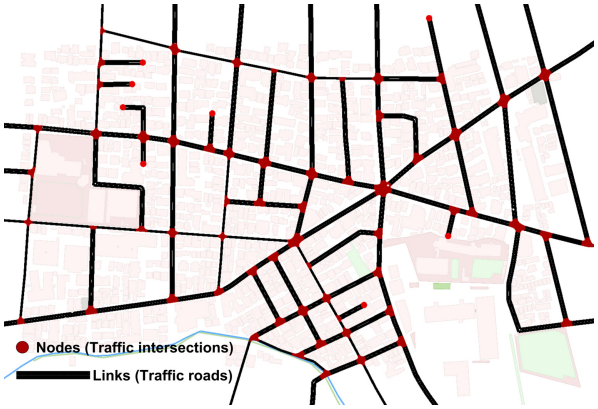


Fig. 9. Traffic network of SUMO simulation in Tokyo, Japan.

TABLE VI
SIMULATION PARAMETERS

Parameters	Description	Value
ΔT	Sampling time	1 s
T_{sim}	Total simulation time	600 s
M	Amount of nodes	90
M_{link}	Amount of links	504
N_{vel}	Amount of vehicles	300
N_{user}	Amount of CAV users	10 - 100
P_{user}	Proportion of CAV users	3.3 - 33.3%
E	Amount of traffic events	0 - 10

tocols and deploying decentralized communication networks, such as blockchain technology, *(ii)* expand the variety of sensors and integrate sensor fusion techniques to provide a more comprehensive and fail-safe approach to environmental perception, *(iii)* explore the use of big traffic data analytics and deploy advanced machine learning algorithms to analyze vast amounts of traffic data more efficiently and accurately, and *(iv)* develop more sophisticated decision-making models that consider a wider range of variables, such as road surface and weather conditions.

V. LARGE-SCALE TRAFFIC SIMULATION AND SYSTEM EVALUATION

To evaluate the effectiveness of proposed CAV navigation system and crucial KPIs of implemented SMDT platform, a set of simulations and experiments are conducted. SUMO is utilized as a large-scale traffic simulator, where we can import test areas from the open street map (OSM), then generate random vehicles and control them with TraCI4Matlab [49]. Subsequently, we can deploy the cooperative event-triggered route planning algorithm in MATLAB to realize real-time path planning for CAV users. Finally, we empirically measured several metrics to validate whether our implemented platform meets the requirements posed by the CAV navigation system.

A. Efficiency and Safety Improved by CAV Navigation System

In SUMO-based traffic simulation, we import an urban traffic area located in Tokyo as illustrated in Fig. 9. The speed limit for each road remains consistent with the default values in OSM, and every intersection is designed to be non-signalized. During the real-time simulation process, at

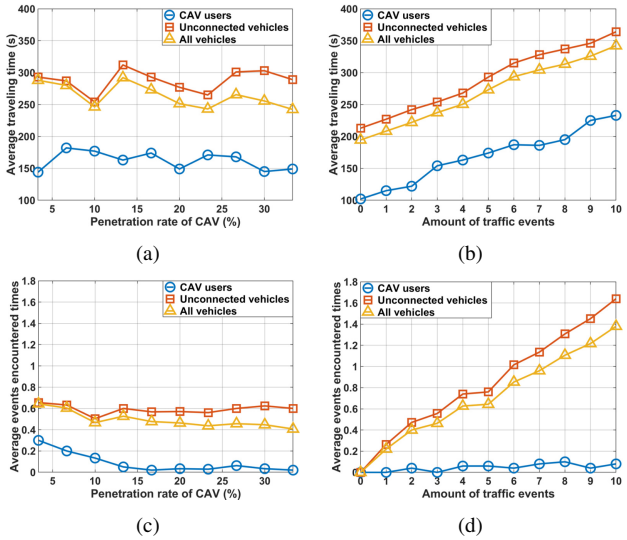


Fig. 10. Simulation results of traffic efficiency and road safety: (a) average travel time with increasing penetration rate of CAV P_{user} ($E = 5$), (b) average travel time with increasing number of traffic events E (Puser $= 16.7\%$), (c) average event encountered times with increasing penetration rate of CAV P_{user} ($E = 5$), (d) average event encountered times with increasing number of traffic events E ($P_{\text{user}} = 16.7\%$).

each sampling time, a random number of vehicles enter the traffic network with both their origins and destinations being randomized. A certain proportion of these vehicles are designated as CAV users, for whom we provide real-time route planning services. In contrast, the remaining vehicles are classified as unconnected vehicles. Lacking dynamic traffic information, they opt for the shortest path between their origin-destination (OD) pairs. Throughout the simulation, a certain number of traffic events, i.e., *pedestrian gathering* and *traffic accidents*, occur at random places in the traffic network. Table VI summarizes the main parameters used in this simulation.

Fig. 10 shows the improved traffic efficiency and road safety with our CAV navigation system. We employ the frequency of encounters with traffic events as an evaluation metric for road safety since such extreme events could pose latent threats and safety concerns for vehicles passing by. In Fig. 10(a), As the number of events increases, both CAV users and unconnected vehicles experience longer driving time to traverse the network. This is predominantly due to that congestion caused by traffic events might propagate to other road segments. Nevertheless, the travel time for CAV users is significantly smaller than that of unconnected vehicles. In Fig. 10(b), it can be observed that, given a constant number of events, with the number of CAV users increasing, the efficiency of CAV users cannot be improved. However, the travel time for the overall traffic consistently shows a declining trend, where some fluctuations are attributed to the random selection of OD pairs and the random occurrence locations of traffic events. As can be discerned from Fig. 10(c), with the occurrence of an increasing number of events, it can be seen that the frequency with which CAV users encounter these events does not show an upward trend. This suggests that the *event-triggered mechanism* in our route planning strategy is effective in assisting users to circumvent such events. As

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3368109

TABLE VII
COMMUNICATION REQUIREMENTS AND PERFORMANCES

	Max. (ms)	Min. (ms)	Mean (ms)
DT modeling process			
I2C comm.	1.74	1.10	1.37
Edge comp. (detection)	153.41	70.01	106.23
T_{dt} Total Max.: 155.15 ms			
Route planning service			
V2C comm.	42.13	20.16	32.30
Cloud comp. (traffic monitoring)	56.29	42.72	45.07
Cloud comp. (route planning)	201.07	173.27	183.68
Edge comp. (NDT localization)	10.13	2.56	6.14
Edge comp. (route loading)	500.97	501.35	501.18
T_{svc} Total Max.: 810.59$<$ 0.164 $v_{\text{max}} = 911.11$ ms			

depicted in Fig. 10(d), with an increasing penetration rate of CAV users, the safety of both CAV users and the overall traffic can be improved, manifesting as a reduction in the frequency of encounters with traffic events.

B. System Evaluation Considering Communication Issues

In DT modeling, the packet delivery rate (PDR) for communication from RSUs to the cloud server is very critical and important, since significant packet loss may lead to incorrect planning decisions. When only processed sensing data (i.e., detection and tracking results) are transmitted, the PDR can approach 100%. However, when simultaneously uploading other levels of data, such as raw LiDAR point clouds, the PDR slightly decreases to approximately 99.53%, which still satisfies the reliability criterion for SSMS, which mandates a PDR greater than 95%. As for the process of providing route planning service, V2C communication is built upon the HTTP protocols. Since HTTP operates atop the Transmission Control Protocol (TCP) at the application layer and TCP ensures data delivery through error detection and retransmissions, the reliability of data transfer under the HTTP protocol can be ensured in stable network conditions.

Some crucial communication and computational latency existing in our system are also measured, as summarized in Table VII. It can be found that both the maximum One-way I2C (1.74 ms) and V2C communication latency (42.1 ms) can meet the requirements proposed by 3GPP, which are 82.6% below the threshold of the max E2E latency for SSMS (less than 10 ms), and 57.9% below the threshold for information sharing (less than 100 ms).

The total latency for *traffic DT modeling* T_{dt} is primarily constituted by the latency derived from edge computing T_{dec} and the one-way I2C communication T_{I2C}, both of which are measured through experiments. Given that the quantity of objects within the LiDAR's sensing range has a large impact on the detection duration, we have conducted measurements of the latency spanning the initiation of the LiDAR to the computation of the detection outcomes many times at various intervals throughout the day. The edge computing exhibited a maximal latency of 153.41 ms, whereas the overall latency for the *traffic DT modeling* reached a peak of 155.15 ms, according to equation (6).

As for the *route planning service* latency, we separately measure V2C communication delay T_{V2C}, cloud computing time T_{cloud} for both traffic monitoring and route planning, and edge computing time for NDT-based localization and route loading/execution. Only the cloud computational time is measured with SUMO-based large-scale simulation, instead of with PoC experiment, because cloud computing in PoC just selects a route from two options, without searching for the fastest route. Based on equation (8), the maximal total latency of the *route planning service* reaches 810.59 ms, which complies with the requirement of CAV user receiving and executing the routing command prior to entering the intersection area. Additionally, the latency of *traffic DT modeling* is also significantly lower than that of *route planning service*.

In the real-world experiment, the frequency for LiDAR-based object detection is set at 30 Hz, which is regarded as the frequency of *traffic DT modeling*. Route planning, on the other hand, is executed on-demand, triggered when the CAV approaches an intersection within the threshold distance S_{thre}. Considering the update frequency and its associated delays, in extreme cases, the latency for *traffic DT modeling* will reach up to 188.48 ms, which is also much lower than that of *route planning service*. For simulations, the cloud computing capabilities operate with a sampling time of 1 s, which is larger than the measured maximum latency of *route planning service*. This setup ensures that cloud services can be completed in real time within each sampling period, demonstrating the cloud's ability to process critical information promptly and efficiently. Taking into account the three crucial aspects of reliability, latency, and frequency, the traffic DT can be regarded as real-time and effective in the context of CAV navigation system.

VI. CONCLUSION

In this study, we have proposed an SMDT platform to offer cloud services for CAV users. By design, this system integrates cloud and edge computing for the establishment of traffic DT. Based on the SMDT platform, we have designed a CAV navigation system that can utilize real-time traffic data and plan routes for CAVs, helping them circumvent dynamic traffic events and improving their efficiency and road safety. In our Smart Mobility Field, we have demonstrated that the designed SMDT-based CAV navigation system can realize effective and real-time navigation in response to traffic events. Future work will aim to expand this platform city-wide, incorporating advanced predictive algorithms for enhanced traffic modeling and exploring its integration within smart city frameworks.

REFERENCES

[1] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital twin in manufacturing: A categorical literature review and classification," *Ifac-PapersOnline*, vol. 51, no. 11, pp. 1016–1022, 2018.

[2] E. VanDerHorn and S. Mahadevan, "Digital twin: Generalization, characterization and implementation," *Decision support systems*, vol. 145, p. 113524, 2021.

[3] F. Tao, H. Zhang, A. Liu, and A. Y. Nee, "Digital twin in industry: State-of-the-art," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2405–2415, 2018.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3368109

[4] C. K. Thomas, W. Saad, and Y. Xiao, "Causal semantic communication for digital twins: A generalizable imitation learning approach," *IEEE Journal on Selected Areas in Information Theory*, vol. 4, pp. 698–717, 2023.

[5] A. Rudskoy, I. Ilin, and A. Prokhorov, "Digital twins in the intelligent transport systems," *Transportation Research Procedia*, vol. 54, pp. 927–935, 2021.

[6] Digital twin market size, share & trends analysis report by end-use (manufacturing, agriculture), by solution (component, process, system), by region, and segment forecasts, 2023 - 2030. [Online]. Available: https://www.grandviewresearch.com/industry-analysis/digital-twin-market

[7] O. Hashash, C. Chaccour, W. Saad, T. Yu, K. Sakaguchi, and M. Debbah, "The seven worlds and experiences of the wireless metaverse: Challenges and opportunities," *arXiv preprint arXiv:2304.10282*, 2023.

[8] T. Yu, Z. Li, K. Sakaguchi, O. Hashash, W. Saad, and M. Debbah, "Internet of federated digital twins (iofdt): Connecting twins beyond borders for society 5.0," *arXiv preprint arXiv:2312.06432*, 2023.

[9] S. Han, F.-Y. Wang, G. Luo, L. Li, and F. Qu, "Parallel surfaces: Service-oriented V2X communications for autonomous vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 11, pp. 4536–4545, 2023.

[10] J. Yang, F. Lin, C. Chakraborty, K. Yu, Z. Guo, A.-T. Nguyen, and J. J. P. C. Rodrigues, "A parallel intelligence-driven resource scheduling scheme for digital twins-based intelligent vehicular systems," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 4, pp. 2770–2785, 2023.

[11] K. Sakaguchi, R. Fukatsu, T. Yu, E. Fukuda, K. Mahler, R. Heath, T. Fujii, K. Takahashi, A. Khoryaev, S. Nagata *et al.*, "Towards mmWave V2X in 5G and beyond to support automated driving," *IEICE Transactions on Communications*, vol. 104, no. 6, pp. 587–603, 2021.

[12] Z. Wang, R. Gupta, K. Han, H. Wang, A. Ganlath, N. Ammar, and P. Tiwari, "Mobility digital twin: Concept, architecture, case study, and future challenges," *IEEE Internet of Things Journal*, vol. 9, no. 18, pp. 17 452–17 467, 2022.

[13] L. Hobert, A. Festag, I. Llatser, L. Altomare, F. Visintainer, and A. Kovacs, "Enhancements of V2X communication in support of cooperative autonomous driving," *IEEE Communications Magazine*, vol. 53, no. 12, pp. 64–70, 2015.

[14] O. Hashash, C. Chaccour, W. Saad, K. Sakaguchi, and T. Yu, "Towards a decentralized metaverse: Synchronized orchestration of digital twins and sub-metaverses," in *ICC 2023-IEEE International Conference on Communications*, 2023, pp. 1905–1910.

[15] X. Hu, S. Li, T. Huang, B. Tang, R. Huai, and L. Chen, "How simulation helps autonomous driving: A survey of sim2real, digital twins, and parallel intelligence," *IEEE Transactions on Intelligent Vehicles*, pp. 1–20, 2023.

[16] Y. Lin, X. Na, D. Wang, X. Dai, and F.-Y. Wang, "Mobility 5.0: Smart logistics and transportation services in cyber-physical-social systems," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 6, pp. 3527–3532, 2023.

[17] Z. Zheng, X. Han, X. Xia, L. Gao, H. Xiang, and J. Ma, "Opencda-ros: Enabling seamless integration of simulation and real-world cooperative driving automation," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 7, pp. 3775–3780, 2023.

[18] Y. Ma, R. Du, A. Abdelraouf, K. Han, R. Gupta, and Z. Wang, "Driver digital twin for online recognition of distracted driving behaviors," *IEEE Transactions on Intelligent Vehicles*, pp. 1–13, 2024.

[19] Z. Wang, O. Zheng, L. Li, M. Abdel-Aty, C. Cruz-Neira, and Z. Islam, "Towards next generation of pedestrian and connected vehicle in-the-loop research: A digital twin co-simulation framework," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 4, pp. 2674–2683, 2023.

[20] Y. Liang, Z. Yin, L. Nie, and Y. Ba, "Shared steering control with predictive risk field enabled by digital twin," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 5, pp. 3256–3269, 2023.

[21] W. Yue, C. Li, G. Mao, N. Cheng, and D. Zhou, "Evolution of road traffic congestion control: A survey from perspective of sensing, communication, and computation," *China Communications*, vol. 18, no. 12, pp. 151–177, 2021.

[22] Y. Cui, H. Xu, J. Wu, Y. Sun, and J. Zhao, "Automatic vehicle tracking with roadside LiDAR data for the connected-vehicles system," *IEEE Intelligent Systems*, vol. 34, no. 3, pp. 44–51, 2019.

[23] L. Wang, X. Zhang, Z. Song, J. Bi, G. Zhang, H. Wei, L. Tang, L. Yang, J. Li, C. Jia, and L. Zhao, "Multi-modal 3D object detection in autonomous driving: A survey and taxonomy," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 7, pp. 3781–3798, 2023.

[24] T. Degrande, S. Van den Eynde, F. Vannieuwenborg, D. Colle, and S. Verbrugge, "C-ITS road-side unit deployment on highways with ITS road-side systems: A techno-economic approach," *IET Intelligent Transport Systems*, vol. 15, no. 7, pp. 863–874, 2021.

[25] V. Tihanyi, A. Rövid, V. Remeli, Z. Vincze, M. Csonthó, Z. Pethő, M. Szalai, B. Varga, A. Khalil, and Z. Szalay, "Towards cooperative perception services for ITS: Digital twin in the automotive edge cloud," *Energies*, vol. 14, no. 18, p. 5930, 2021.

[26] Y. Guo, K. Zou, S. Chen, F. Yuan, and F. Yu, "3D digital twin of intelligent transportation system based on road-side sensing," in *Journal of Physics: Conference Series*, vol. 2083, no. 3. IOP Publishing, 2021, p. 032022.

[27] J. Barrachina, P. Garrido, M. Fogue, F. J. Martinez, J.-C. Cano, C. T. Calafate, and P. Manzoni, "Road side unit deployment: A density-based approach," *IEEE Intelligent Transportation Systems Magazine*, vol. 5, no. 3, pp. 30–39, 2013.

[28] O. K. Tonguz and W. Viriyasitavat, "Cars as roadside units: a self-organizing network solution," *IEEE Communications Magazine*, vol. 51, no. 12, pp. 112–120, 2013.

[29] R. Razdan, M. İ. Akbaş, R. Sell, M. Bellone, M. Menase, and M. Malayjerdi, "Polyverif: An open-source environment for autonomous vehicle validation and verification research acceleration," *IEEE Access*, vol. 11, pp. 28 343–28 354, 2023.

[30] Y. Liu, B. Sun, Y. Tian, X. Wang, Y. Zhu, R. Huai, and Y. Shen, "Software-defined active LiDARs for autonomous driving: A parallel intelligence-based adaptive model," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 8, pp. 4047–4056, 2023.

[31] T. Zeng, O. Semiari, M. Chen, W. Saad, and M. Bennis, "Federated learning on the road autonomous controller design for connected and autonomous vehicles," *IEEE Transactions on Wireless Communications*, vol. 21, no. 12, pp. 10 407–10 423, 2022.

[32] Y. He, B. Wu, Z. Dong, J. Wan, and W. Shi, "Towards C-V2X enabled collaborative autonomous driving," *IEEE Transactions on Vehicular Technology*, 2023.

[33] O. S. Oubbati, M. Atiquzzaman, P. Lorenz, A. Baz, and H. Alhakami, "Search: An SDN-enabled approach for vehicle path-planning," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 14 523–14 536, 2020.

[34] T. Wágner, T. Ormándi, T. Tettamanti, and I. Varga, "SPaT/MAP V2X communication between traffic light and vehicles and a realization with digital twin," *Computers and Electrical Engineering*, vol. 106, p. 108560, 2023.

[35] Z. Li, K. Wang, T. Yu, and K. Sakaguchi, "Het-SDVN: SDN-based radio resource management of heterogeneous V2X for cooperative perception," *IEEE Access*, vol. 11, pp. 76 255–76 268, 2023.

[36] Q. Liu, T. Han, J. Xie, and B. Kim, "Real-time dynamic map with crowdsourcing vehicles in edge computing," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 4, pp. 2810–2820, 2023.

[37] Z. Wang, K. Han, and P. Tiwari, "Digital twin-assisted cooperative driving at non-signalized intersections," *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 2, pp. 198–209, 2022.

[38] J. Dong, Q. Xu, J. Wang, C. Yang, M. Cai, C. Chen, Y. Liu, J. Wang, and K. Li, "Mixed cloud control testbed: Validating vehicle-road-cloud integration via mixed digital twin," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 4, pp. 2723–2736, 2023.

[39] M. H. Cheung, F. Hou, V. W. Wong, and J. Huang, "DORA: Dynamic optimal random access for vehicle-to-roadside communications," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 4, pp. 792–803, 2012.

[40] M. Xing, J. He, and L. Cai, "Maximum-utility scheduling for multimedia transmission in drive-thru internet," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 4, pp. 2649–2658, 2015.

[41] Institute of Transportation Engineers. Technical Council Committee 4A-16, "Determining vehicle change intervals: A proposed recommended practice," *Institute of Transportation Engineers*, 1985.

[42] M. H. C. Garcia, A. Molina-Galan, M. Boban, J. Gozalvez, B. Coll-Perales, T. Şahin, and A. Kousaridas, "A tutorial on 5G NR V2X communications," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1972–2026, 2021.

[43] 3GPP, "Study on enhancement of 3GPP support for 5G V2X services (Release 16)," *Technical Report 22.886 V16.2.0*, 2018.

[44] Autoware Universe Documentation. [Online]. Available: https://autowarefoundation.github.io/autoware.universe/main/.

[45] T. Yin, X. Zhou, and P. Krahenbuhl, "Center-based 3D object detection and tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 11 784–11 793.

[46] Autoware Multi-object Tracker. [Online]. Available: https://autowarefoundation.github.io/autoware.universe/main/perception/multi_object_tracker/.

[47] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "Sumo–simulation of urban mobility: an overview," in *Proceedings of SIMUL*

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3368109

2011, The Third International Conference on Advances in System Simulation. ThinkMind, 2011.

[48] UC-win/Road simulator. [Online]. Available: https://www.forum8.co.jp/english/uc-win/ucwin-road-e1.htm.

[49] A. F. Acosta, J. E. Espinosa, and J. Espinosa, "Traci4matlab: Enabling the integration of the sumo road traffic simulator and matlab® through a software re-engineering process," in *Modeling Mobility with Open Data: 2nd SUMO Conference 2014 Berlin, Germany, May 15-16, 2014*. Springer, 2015, pp. 155–170.



Kui Wang (Student Member, IEEE) received the B.E. degree in mechanical engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 2018, and the M.E. degree in vehicle engineering from the KTH Royal Institute of Technology, Sweden, in 2021. He is currently working toward the Ph.D. degree in the Department of Electrical and Electronic Engineering, Tokyo Institute of Technology, Tokyo, Japan. His current research interests include smart mobility, autonomous driving, digital twin, and federated learning.



Zongdian Li (Member, IEEE) received the B.E. degree in communication engineering from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2018, and the M.E. and Ph.D. degrees in electrical and electronic engineering from Tokyo Institute of Technology (Tokyo Tech), Tokyo, Japan, in 2020 and 2023, respectively. Since October 2023, he has been an Assistant Professor with the Department of Electrical and Electronic Engineering, School of Engineering, Tokyo Tech. His research interests include software-defined networking (SDN), vehicle-to-everything (V2X) communications, digital twins, and autonomous driving. He is a member of IEICE.



Kazuma Nonomura (Student Member, IEEE) received the B.E. degree in electrical and electronic engineering from the Tokyo Institute of Technology, Tokyo, Japan, in 2022. He is currently working toward the M.E. degree in the Department of Electrical and Electronic Engineering, Tokyo Institute of Technology, Tokyo, Japan. His current research interests include smart mobility, autonomous driving, digital twin, and reinforcement learning.



Tao Yu (Member, IEEE) received the M.E. degree from the Communication University of China (CUC), Beijing, China in 2010, and the Dr.Eng. degree from the Tokyo Institute of Technology (Tokyo Tech), Tokyo, Japan in 2017. From 2017 to 2022, he worked as a postdoctoral researcher in Sakaguchi Lab, Dept. Electr. Electron. Eng., Tokyo Tech. From 2022, he joined the Tokyo Tech Academy for Super Smart Society where he is currently a specially appointed associate professor. His research interests include UAV communication, V2X, sensor networks, localization, antenna, smart mobility, and building energy management. He is member of IEEE and IEICE.



Kei Sakaguchi (Senior Member, IEEE) received the M.E. degree in information processing from Tokyo Institute Technology, Tokyo, Japan, in 1998, and the Ph.D. degree in electrical and electronics engineering from Tokyo Institute Technology, Tokyo, Japan, in 2006. He is currently working with the Tokyo Institute of Technology in Japan, as the Dean with Tokyo Tech Academy for Super Smart Society and as a Professor with the School of Engineering. At the same time, he is working for oRo Co., Ltd. in Japan as the outside Director. His current research interests include 5G cellular networks, millimeter-wave communications, wireless energy transmission, V2X for automated driving, and super smart society. He was the recipient of the Outstanding Paper Awards from SDR Forum and IEICE, in 2004 and 2005, respectively, and three Best Paper Awards from IEICE communication society in 2012, 2013, and 2015. He was also the recipient of the Tutorial Paper Award from IEICE communication society in 2006. He is a fellow of IEICE.



Omar Hashash (Student Member, IEEE) received his B.E. in Communications and Electronics Engineering from Beirut Arab University, Lebanon in 2019 and his M.E. in Electrical and Computer Engineering from the American University of Beirut, Lebanon in 2021. He is currently a Ph.D. student at the Electrical and Computer Engineering Department at Virginia Tech. His research interests include 6G wireless networks, digital twins, metaverse, edge intelligence, and generalizable AI. He has recently published one of the first works that explore the synergy between wireless, computing, and AI techniques that can come together to support massive digital twinning of physical systems in the metaverse.



Walid Saad (Fellow, IEEE) received his Ph.D degree from the University of Oslo, Norway in 2010. He is currently a Professor at the Department of Electrical and Computer Engineering at Virginia Tech, where he leads the Network sciEnce, Wireless, and Security (NEWS) laboratory. His research interests include wireless networks (5G/6G/beyond), machine learning, game theory, security, UAVs, semantic communications, cyber-physical systems, and network science. Dr. Saad is a Fellow of the IEEE. He is also the recipient of the NSF CAREER award in 2013, the AFOSR summer faculty fellowship in 2014, and the Young Investigator Award from the Office of Naval Research (ONR) in 2015. He was the (co-)author of twelve conference best paper awards at IEEE WiOpt in 2009, ICIMP in 2010, IEEE WCNC in 2012, IEEE PIMRC in 2015, IEEE SmartGridComm in 2015, EuCNC in 2017, IEEE GLOBECOM (2018 and 2020), IFIP NTMS in 2019, IEEE ICC (2020 and 2022), and IEEE QCE in 2023. He is the recipient of the 2015 and 2022 Fred W. Ellersick Prize from the IEEE Communications Society, of the IEEE Communications Society Marconi Prize Award in 2023, and of the IEEE Communications Society Award for Advances in Communication in 2023. He was also a co-author of the papers that received the IEEE Communications Society Young Author Best Paper award in 2019, 2021, and 2023. Other recognitions include the 2017 IEEE ComSoc Best Young Professional in Academia award, the 2018 IEEE ComSoc Radio Communications Committee Early Achievement Award, and the 2019 IEEE ComSoc Communication Theory Technical Committee Early Achievement Award. From 2015-2017, Dr. Saad was named the Stephen O. Lane Junior Faculty Fellow at Virginia Tech and, in 2017, he was named College of Engineering Faculty Fellow. He received the Dean's award for Research Excellence from Virginia Tech in 2019. He was also an IEEE Distinguished Lecturer in 2019-2020. He has been annually listed in the Clarivate Web of Science Highly Cited Researcher List since 2019. He currently serves as an Area Editor for the IEEE Transactions on Network Science and Engineering and the IEEE Transactions on Communications. He is the Editor-in-Chief for the IEEE Transactions on Machine Learning in Communications and Networking.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/