Web-Based Car Control and Monitoring for Safe Driving of Autonomous Vehicles

1st Junhee Kwon
Computer Science & Engineering
Sungkyunkwan University
Suwon, Republic of Korea
juun9714@skku.edu

2nd Bien Aime Mugabarigira
Electrical & Computer Engineering
Sungkyunkwan University
Suwon, Republic of Korea
bienaaime@skku.edu

3rd Jaehoon (Paul) Jeong
Computer Science & Engineering
Sungkyunkwan University
Suwon, Republic of Korea
pauljeong@skku.edu

2022 International Conference on Information Networking (ICOIN) | 978-1-6654-1332-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICOIN53446.2022.9687175

***Abstract*—The future of vehicular transportation will rely on vehicular communication to facilitate a safe, reliable and comfortable driving. The recent research has defined the vehicular communication protocols enabling vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), vehicle-to-pedestrian (V2P) and vehicle-to-everything (V2X) communications. In this paper, we design a web-based car monitoring that enables vehicles in vehicular network to exchange their mobility information to achieve quick road awareness using W3C Vehicle Information Service Specification (VISS) and Vehicle Signal Specification (VSS). This monitoring takes advantage of the vehicle network infrastructure (i.e., Traffic Control Center (TTC)) to control and monitor road mobility thus enhancing the safety of autonomous driving vehicles. We conducted the experimental testing to validate the proposed architecture on the AION R1 robot car. We designed the analysis for accessing R1 robot car's database via Django server to get the event observed in the road. In the result, V2I communication showed a delay of 0.055 seconds and V2V communication showed an average delay of 0.0015 seconds, giving enough time for distant vehicles to proactively react on it.**

***Index Terms*—VANET, VISS, VSS, vehicle mobility information, connected vehicle, accident prevention.**

I. Introduction

Recently, the interest in autonomous driving research has increased due to their promises to revolutionize the safety of vehicular transportation. Autonomous driving is made possible by checking and determining the situation of the road through data from a car's on-board sensors. An autonomous car is intelligent enough to assess the road condition to act and react towards a safe driving.

Autonomous cars' sensing data are obtained in two ways. The first is the observation of a vehicle's various sensors. The second is the data from vehicular networks. An alternative method consists of receiving data from other nearby vehicles or from the infrastructure on the road. Three communication methods are commonly known in vehicular networks: vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), and the vehicle-to-everything (V2X) communications [1]. The service-enabled vehicular communication enables vehicles to be informed of the driving environment in a wider scope.

An autonomous vehicle's environment perception that is only based on the information obtained through its own sensors is limited to a sensor's line-of-sight. Therefore, it is not reliable enough for safe driving. Sensing should be a real-time activity to respond to road events in a timely way. Sensors may have inaccurate measurement when driving in an environment with physical obstacles and it is hard to predict situations at a distance. On the other hand, when autonomous driving is performed with data obtained based on communication through either V2I or V2V, it is possible to get data on the road even for a long physical distance or obstacles. However, this wireless communication-based method has a disadvantage that there may be packet loss or delay during communication [2]. The use of this V2X technology is a good approach to give a wide range of cognition, therefore it can cope with unexpected situations by receiving the communicated information from nearby vehicles.

In fact, vehicular ad hoc networks (VANET) can be constructed when vehicles are driving closely, enabling them to exchange messages among them. Thus, when their inter-distances grow large, the vehicles cannot communicate with each other and this grows the risk of loosing the awareness of the road. Another issue of VANET based safety message exchange is the partitioning and merging of vehicular networks. In such a scenario, to communicate with distant vehicles should rely on carry and forward communication [3]. The processing of data with multiple relay nodes increases their delivery delay to destination and, it also increases the probability of packet loss. With a long road segment, there will also be many disconnected VANETs due to disconnection caused by driving out of the current packet carrier's communication range. Therefore, we need an alternative plan to cover the mentioned deficiencies of the V2V communication in vehicular networks.

The V2I-communication increases the possibility of communication coverage in wider range. Another benefit of the infrastructure is the communication security improvement. Basic network functions such as packet forwarding, routing, and network management are possible when realized by the fixed infrastructure nodes in network. However in VANET, there are not fixed nodes. The nodes of VANET are not reliable enough to carry the critical network functions. [4]. The V2X combines the V2I and V2V benefits, thus resulting in an improved and secured packet communication on the road. A Traffic Control Center (TTC) is a centralized vehicular infrastructure that manages and controls road traffics. A recent research proposed the communication, management and processing of safety driving controls though Web-VANET (WVANET) [5], [6].

The WVANET infrastructure node on the road can provide wider scope than VANET [6]. By the use of a web-based message exchange method via an available network infrastructure, it disseminates the safety control messages among vehicles. With a large number of the vehicles on the road, WVANET-based communication is faster than the data communication through V2V. Another benefit is that a web-based node can enable security options, thus securing the vehicular communication.

Inspired by the web-based vehicular safety, this paper analyzes the exchange of messages between infrastructure and vehicles as illustrated in Fig. 1. In our structure, infrastructure requests the VMI to R1 Robot car via REST API as a client. Then R1 robot car responses to client's request with its VMI data as a server based on Django. The vehicles are capable of communicating with neighboring vehicles in 25m without packet loss. Through Django-based server, we propose the use of Vehicle Information Service Specification (VISS) and Vehicle Signal Specifications (VSS) [7], to construct a vehicular web. We conducted the experiment and checked the delay of the exchange of safety messages such as Cooperative Context Message (CCM) and Emergency Context Messages [8] through both V2V and V2I. The experiment performance results show that it is faster to deliver safety data through V2V than V2I. However, V2V allows vehicles to communicate with each other even in an environment where the network is unstable or not established. On the other hand, V2I allows vehicles can get the information on the road even there is no neighboring vehicles. As these two communication methods have different use cases, the results of these would not be compared.

In this paper, we suggest a more robust vehicular network on the road. We combined the V2V and V2I structures to make safer vehicle communication. Through a Django server on R1 robot car, we set up an infrastructure network on the real road and we utilize two robot cars for vehicles' role. These nodes keep exchanging the safety messages to alert back and forth the road condition in a vehicular network.

The remainder of this paper is organized as follows. Section II summarizes the state-of-the-art of web-based safe driving. Section III describes our proposed web based car control system design. Section IV shows the evaluation and validation of our proposed mechanism. Lastly, Section V concludes our paper along with the future work.

II. Related Work

The vehicular network-based cars' tracking and safety preserving systems have been an important aspect of the autonomous driving research. This section reviews the protocols and the existing web-based methods proposed by the recent literature.

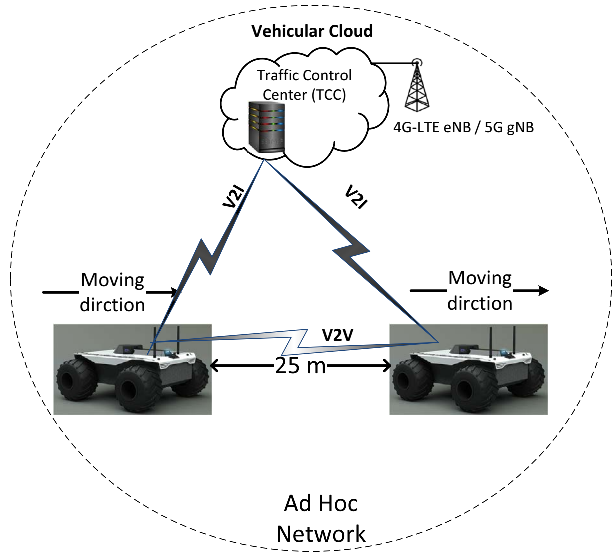

Fig. 1. Vehicular cloud system structure.

The IP wireless access in vehicular environments (IP-WAVE) has designed the necessary protocols for vehicular networks [1], [9]–[11]. The communication methods and settings to use the IPv6 over IEEE 802.11-outside of the context of a basic service set (OCB link) provide basics for vehicular communication [9]. However, the security and privacy of vehicular networks are needed for securing the vehicular communication. [11] describes possible attacks in vehicular networks and their corresponding countermeasures. Vehicular communication in autonomous driving also enables the driving safety enhancement. [1] describes a light-weight message exchange among vehicles to enable their cooperation for safe driving. Context-Aware Navigation Protocol (CNP) [10] specified the neighbour discovery (ND) message option such as cooperation context message (CCM) and emergency context message (ECM) for road situation awareness. For attaining remote driving situation diagnosis and management services, the web is a better alternative [12]–[15].

An intelligent vehicle system that remotely diagnoses and manages vehicles was proposed in [12]. An in-vehicle sensor network is capable of status detection, control and communication, thus offering an interactive service in vehicle mobility environments. Fong et. al developed a system for users to access vehicle teleoperation [13]. Through sensors' fusion, this web-based system enables a collaborative control. It has been demonstrated that a web platform can locally and remotely control and monitor nodes in a network [14]. This ability can reach a driving car by controlling its ignition, speed and steering systems.

The spider-web-like for emergency data transmission in VANET using dynamic multi-priority packet scheduling was proposed in [16]. It classifies a packet into three priorities which are regular and emergency messages to enqueue the received packets. The web-VANET (WVANET) provides a communication model that uses web signals to deliver vehicular information [6]. It offers both flexible communication and service discovery in a connected vehicular network. It enables

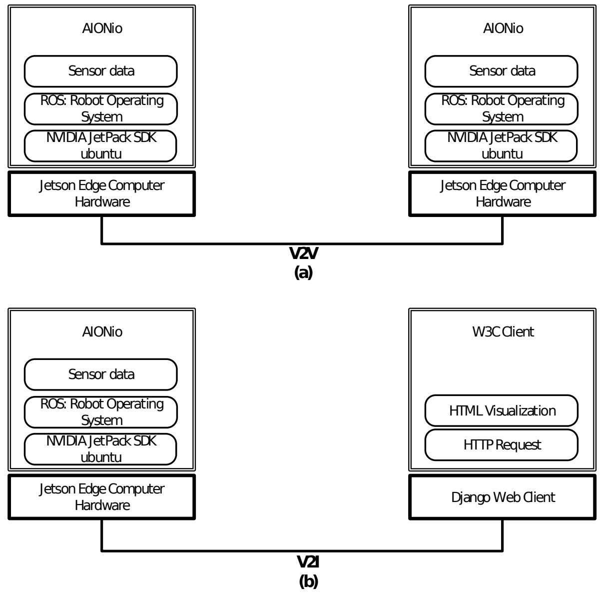

Fig. 2. A web-based system structure.

single hop communication with very less link disconnection and small transmission time which avoid data losses. A cloud database enables vehicle mobility monitoring and control in autonomous vehicles. Chen. et. al argued that spider-web-based vehicular networks can remedy the challenge of reliable route establishment for delivering packets in VANET [17]. A connection-quality model and transmission-latency model find a feasible route for a vehicle's navigation system. Joe et. al proposed an architecture that WiMAX technology establishes the communication among vehicles [18]. In their design, a single WiMAX tower located miles away can offer a web service through its signal strength accessible to road-side units (RSUs).

The web in VANET makes use of web signals to dispatch safety alerts and commands to vehicles [6]. A transmission mechanism for emergency data (TMED) has overcome the limitations of computation complexity of web-like vehicle systems by greedily forwarding dynamic multi priority message queues [16]. Sensors such as global positioning system (GPS) and global system for mobile (GSM) were used to for accident prevention in vehicle web systems [19]. However, the efficiency of those systems and impact of V2I communication on the driving safety were not treated by vehicle control systems. In this paper, we tested the vehicular web for vehicular safety message exchanges. We analyze both V2I and V2V communication through AION R1 robot cars [20].

III. Design

This section discusses the architecture and working flow of our proposed web-based scheme to control and monitor the safety of autonomous vehicles.

A. A Web-Based Car Control System Architecture

The vehicular network design that we proposed considers inter-vehicular communications (V2V) and the vehicle-to-infrastructure communication (V2I), as illustrated in Fig. 1.

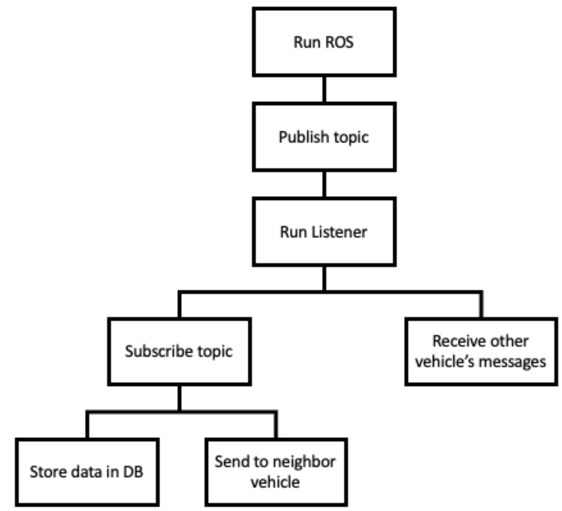

Fig. 3. Vehicular Web Client's work-flow diagram.

A car is a mobile node with the sensing, processing and computation capacity. A car issues its own VMI and provides its information if an infrastructure node requests it. VMI can be information about car status such as GPS, battery, temperature, and speed of the car.

In the design proposed by this paper, TTC plays a role of a web client. It performs of collecting the communicated VMI (e.g., speed, position, direction and action), processing, maintaining and delivering it to cars within its communication range to prevent and avoid accidents. A car interior system has a robot operating system (ROS) [21] that helps to build an algorithm to perceive and communicate data from its sensors (i.e., GPS). Seamlessly, the car serves its sensed data to the neighboring vehicles and infrastructure node (i.e., a traffic control center (TTC)).

B. System Working Flow

The working flow of the proposed web-enabled control and monitoring for the autonomous vehicles' driving safety consists of the environment sensing, processing sensing data and sensing data delivery to a neighboring vehicle or cloud server. Cars communicate with each other over VANET that they create. A typical car-to-car communication is illustrated by Fig. 2 (a).

Indeed, Fig. 2 (a) shows the AION R1 robot car system structure that we used. A robot car is comprised of sensor, Robot Operating System (ROS) which is a meta OS running on Ubuntu, Ubuntu on NVIDIA Jetpack SDK, and Jetson Edge Compute Hardware including a motor driver. ROS' role is to provide the data gathered from the R1's on-board sensor to the pilot system. Sensor data are provided by making a topic to which the pilot system subscribes by using methods provided by ROS. The car's on-board network interface card (NIC) connects to remote network wirelessly or creates an ad-hoc network. A mounted GPS sensor observes a vehicle's position regularly. The optimised Jetson computer of AION R1 makes it possible to process data in an accelerated way therefore getting timely updates of a vehicle's position.

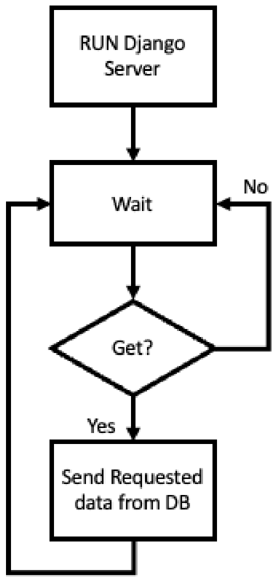

Fig. 4. Vehicular Django web server's work-flow.

The sensed data are shared through a VANET created by robots in a V2V communication fashion. On the other hand, a robot car also provides its data to the TCC through a web-based REST API that we created to portray the V2I communication. A Web client, which serves as an infrastructure node in our design, can manage data comprehensively and make a useful decision by utilizing accumulated data from the vehicles. Combining both V2V and V2I communications improves the safety of a vehicular network.

Besides, in our proposed V2I, a web client receives data packets from the vehicles and deploys the collected data in an integrated manner. Fig. 2 (b) illustrates a web client node's structure (names W3C Client) running on same network. We assumed that web client and R1 robot car are in same network. We used Django stack to make a server-side application on R1 robot car as it supports a lot of useful libraries. A vehicle sends its packet to a remote client by using REST API. REST API is the most popular methods in web communication and exchanges the data without an established connection.

Fig. 3 shows the work-flow diagram of a robot car. A car runs the ROS that publishes a topic including the VMI of the robot car. Then, the robot car runs the listener program that subscribes to the topic that issues the sensor data of robot car. At the same time, a listener stores the subscribed data on its storage with VSS format. It provides subscribed data to a web client and neighboring vehicles. It also receives the neighboring vehicles' data. Fig. 4 describes the work-flow of the Django server on R1 robot car. Server on R1 robot car runs the Django server and waits for the request from the web client. It can handle GET method requesting data to the Django server. If a robot car responds its VMI to the client, the client shows received data on its web page visually. In the next section, we describe the performance of this proposed design.

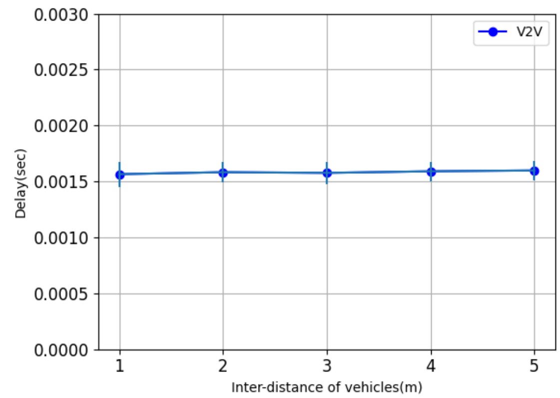

Fig. 5. Average Delay according to Distance between Vehicles

IV. Performance Evaluation

This section describes the experiment environment and the performance results of a web-based driving safety system.

A. Experiment

We conducted an experiment of two AION R1 robot cars with Django server and an web client. We evaluated the transmission delay of both V2V and V2I communications. The V2V results use a 95% confidence interval.

We carried out the experiment by evaluating the V2V communication delay of two distant vehicles. For each distance between two vehicles that we tested, multiple packets were transmitted. Therefore, we used the average transmission delay as a metric of the V2V communication. The delay is one-way delivery delay from sending of a message till it is received. Due to the robot's system time synchronization, the Round-Trip-Time (RTT) of a message was used to evaluate the one-way packet delay according to Equation (1). RTT is the time difference between sending time and receiving time of the echoed message from the destined receiving car. Therefore, the packet delivery delay is the half of RTT. For each tested distance, we recorded the average delay by collecting messages' delays sent within 60 seconds.

$$D = \frac{RTT}{2}. \quad (1)$$

B. Testing results

Fig. 5 depicts the average transmission delays of the V2V communication. We recorded the average one-way delay for 1 minutes, 10 times per each distance. Horizontal axis means the distance between two vehicles and vertical axis means the average delay of a message for each distance. Also we represented the error bar for each distance. We checked the one-way delay of V2V communication with increasing distance. Regardless of the inter-distance of vehicles, the transmission delay remained constant. The average one-way delay of V2V communication was about 0.0015 seconds.

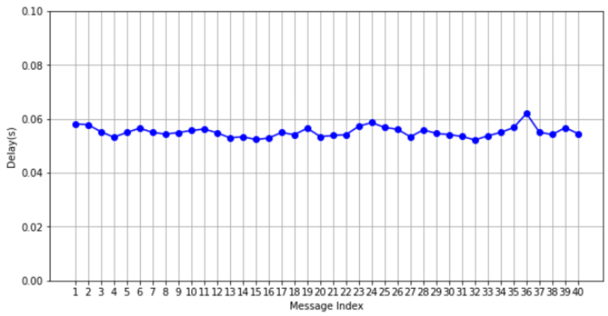

Fig. 6. Delay of Message between Vehicle and Web Client

We implemented a remote web client as an infrastructure node. V2I communication covers a large distance, therefore offering a higher communication scope. In Fig. 6 we investigated the packet transmission delay of a packet to an web client. Horizontal axis means the index of the message that the destination vehicle got from the R1 robot car's server and vertical axis means the delay of a message. For 40 messages we inspected, their transmission delay was about 0.055 seconds. Therefore, neighboring vehicles' VMI may be collected and provided in real time through a fast communication speed of V2I. At the same time, infrastructure node enables efficient decision-making for safe driving on the road by integrating data from multiple vehicles.

Though the delay of the V2I communication is larger than the delay of the V2V communication, the V2I's use case is different from the V2V's use case. V2I and V2V are implemented and used in a form that can compensate for defects in each method. If the network is unstable or not established on the road, it may be difficult for the infrastructure node to collect data from vehicles and provide integrated judgement. In that case, vehicles are designed to cope with this situation by forming an ad-hoc network with each other. On the other hand, V2I communication can make it possible to provide the road's condition when there is no neighboring vehicles.

V. Conclusion

This paper introduced a Web Vehicular Ad Hoc Network (WVANET) and Django server to control and monitor vehicles to attain the safe driving of autonomous vehicles. In this paper, VISS and VSS standards were applied to test whether the Vehicular network's standards under development can be used in actual communication. Through an implementation of V2V and V2I communication among two AION R1 robot cars and web client as infrastructure node, we investigated the V2X-based safety information communication. The testing results show that both V2V and V2I can be used to deliver safety messages in different road situations.

A web-based server can provide security options and encryption method such as HTTPS, therefore securing the vehicular communication. As future work, we will extend the proposed architecture in this paper to secure the autonomous vehicle communication. We will also compare our safety monitoring system to the existing safety monitoring systems in an autonomous vehicle on top of AION R1 robots to assess the quality of service for safe driving.

Acknowledgments

This work was supported by the ITRC (Information Technology Research Center) support program (IITP-2021-2017-0-01633) and the IITP (Institute for Information & Communications Technology Planning & Evaluation) grant (No. 2019-0-01343, Regional strategic industry convergence security core talent training business). Note that Jaehoon (Paul) Jeong is the corresponding author.

References

[1] J. P. Jeong, "IPv6 Wireless Access in Vehicular Environments (IPWAVE): Problem Statement and Use Cases," Internet Engineering Task Force, Internet-Draft draft-ietf-ipwave-vehicular-networking-24, Oct. 2021, work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-ipwave-vehicular-networking-24

[2] I. Parra, H. Corrales, N. Hernández, S. Vigre, D. Llorca, and M. Sotelo, "Performance analysis of vehicle-to-vehicle communications for critical tasks in autonomous driving," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019, pp. 195–200.

[3] J. Jeong, S. Guo, Y. Gu, T. He, and D. H. Du, "Trajectory-based data forwarding for light-traffic vehicular ad hoc networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 5, pp. 743–757, 2010.

[4] P. Michiardi and R. Molva, "Ad hoc networks security," *ST JOURNAL OF SYSTEM RESEARCH*, vol. 4, pp. 1–3, 2003.

[5] H. Liu, T. Qiu, X. Zhou, C. Chen, and N. Chen, "Parking-area-assisted spider-web routing protocol for emergency data in urban vanet," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 971–982, 2019.

[6] M. M. Joe and B. Ramakrishnan, "Review of vehicular ad hoc network communication models including wvanet (web vanet) model and wvanet future research directions," *Wireless networks*, vol. 22, no. 7, pp. 2369–2386, 2016.

[7] W3V. Vehicle information service specification. [Online]. Available: https://www.w3.org/TR/vehicle-information-service/

[8] B. A. Mugabarigira, Y. C. Shen, D. Choe, and J. P. Jeong, "Context-aware navigator for road safety in vehicular cyber-physical systems."

[9] N. Benamar, J. Haerri, J.-H. Lee, T. Ernst, and T. Ernst, "Basic Support for IPv6 Networks Operating Outside the Context of a Basic Service Set over IEEE Std 802.11," RFC 8691, Dec. 2019. [Online]. Available: https://rfc-editor.org/rfc/rfc8691.txt

[10] J. P. Jeong, B. A. Mugabarigira, Y. C. Shen, Z. Xiang, and Z. I. Kim, "Context-Aware Navigation Protocol for IP-Based Vehicular Networks," Internet Engineering Task Force, Internet-Draft draft-jeong-ipwave-context-aware-navigator-04, Aug. 2021, work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-jeong-ipwave-context-aware-navigator-04

[11] J. P. Jeong, Y. C. Shen, and P. Jung-Soo, "Basic Support for Security and Privacy in IP-Based Vehicular Networks," Internet Engineering Task Force, Internet-Draft draft-jeong-ipwave-security-privacy-04, Aug. 2021, work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-jeong-ipwave-security-privacy-04

[12] T.-H. Kim, S.-I. Lee, Y.-D. Lee, and W.-K. Hong, "Design and evaluation of in-vehicle sensor network for web based control," in *13th Annual IEEE International Symposium and Workshop on Engineering of Computer-Based Systems (ECBS'06)*. IEEE, 2006, pp. 10–pp.

[13] T. Fong, C. Thorpe, and C. Baur, "Advanced interfaces for vehicle teleoperation: collaborative control, sensor fusion displays, and web-based tools," in *Vehicle Teleoperation Interfaces Workshop, IEEE International Conference on Robotics and Automation*, no. CONF, 2000.

[14] A. Yazidi, H. Henao, G.-A. Capolino, F. Betin, and F. Filippetti, "A web-based remote laboratory for monitoring and diagnosis of ac electrical machines," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 10, pp. 4950–4959, 2011.

[15] M. Al-Rousan, F. Jalajel, F. Arain, and A. Auf, "Web-based wireless vehicle control system," in *10th IEEE International Conference on Electronics, Circuits and Systems, 2003. ICECS 2003. Proceedings of the 2003*, vol. 1. IEEE, 2003, pp. 388–391.
[16] T. Qiu, X. Wang, C. Chen, M. Atiquzzaman, and L. Liu, "Tmed: A spider-web-like transmission mechanism for emergency data in vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 8682–8694, 2018.
[17] C. Chen, L. Liu, T. Qiu, K. Yang, F. Gong, and H. Song, "Asgr: An artificial spider-web-based geographic routing in heterogeneous vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 5, pp. 1604–1620, 2018.
[18] M. M. Joe and B. Ramakrishnan, "Wvanet: Modelling a novel web based communication architecture for vehicular network," *Wireless personal communications*, vol. 85, no. 4, pp. 1987–2001, 2015.
[19] M. Abdulwaheed and S. MASHOOD, "Vehicle tracking and accident alert system using gps and gsm modules," 2019.
[20] A. Robotics. Aion robotics. [Online]. Available: https://www.aionrobotics.com/
[21] ROS. Robot operating system. [Online]. Available: https://www.ros.org/