

Research Article

High-Capacity Data Collection Platform for Smart Cities Using IEEE 802.11ad-Based Millimeter-Wave V2X Communication

Kosei Nakano ¹, Hiroyuki Motozuka ², Gaius Yao Huang Wee,³ Masataka Irie,¹ Akihiro Egami,¹ Takenori Sakamoto,¹ Koji Takinami,¹ and Kazuaki Takahashi¹

¹Panasonic Industry, Co., Ltd., 600 Saedo-cho, Tsuzuki-ku, Yokohama City 224-8539, Japan

²Panasonic Intellectual Property Management Co., Ltd., 600 Saedo-cho, Tsuzuki-ku, Yokohama City 224-8539, Japan

³Panasonic R&D Center Singapore, Panasonic Asia Pacific Pte. Ltd., Singapore 469332

Correspondence should be addressed to Kosei Nakano; nakano.kosei@jp.panasonic.com

Received 19 April 2022; Accepted 22 June 2022; Published 9 July 2022

Academic Editor: Zahid Khan

Copyright © 2022 Kosei Nakano et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The collection and utilization of huge sensor data from vehicles for visualizing the city is expected to realize various enhanced services for smart cities. A next-generation gigabit vehicle-to-everything (V2X) data collection platform based on 60 GHz millimeter-wave (mmWave) small cell radio access has been proposed in the previous work for enabling efficient and high-capacity data upload from vehicles to the cloud. This paper presents further analysis of the initial link setup delay to evaluate the effectiveness of the fast initial V2X link setup method proposed, which is suitable for 60 GHz communication based on IEEE 802.11ad. This paper also presents the application programming interface (API) design based on the hypertext transport protocol (HTTP) to enable the efficient upload of hundreds of files. Combined with the buffering technique at a multiaccess edge computing (MEC) server, the proposed system successfully utilizes very high bandwidth between a vehicle and a MEC server. The proposed methods achieve a 30 times improvement in delay for establishing the initial link and an 11 times higher average throughput. A prototype system installed in Marysville, Ohio, achieved a peak throughput of 2.8 Gbps and successfully demonstrates the effectiveness of city visualization based on high-capacity data collection and analytics.

1. Introduction

With the recent advances in Advanced Driver Assistance Systems (ADAS) and autonomous driving technologies, vehicles are expected to collect huge amounts of data from onboard sensors such as cameras, sonars, and Global Positioning System (GPS)/Global Navigation Satellite System (GNSS) devices [1, 2]. By analyzing the collected data and visualizing the city, various services for smart cities can be realized, such as enhanced traffic control and expedited emergency response [3]. Demonstrations which utilize such collected data have been on the rise in various cities and countries [4], and efficient data collection of real-time high-resolution data and massive amounts of delay-tolerant data [5] via wireless communication from vehicles in the city is becoming increasingly relevant.

Existing vehicle-to-everything (V2X) communication technologies such as IEEE 802.11p-based Dedicated Short-Range Communication (DSRC) [6] and Cellular-V2X (C-V2X) [7] are not suitable for such high-capacity data collection due to their limited bandwidth. Recently, millimeter-wave (mmWave) wireless communication has been increasingly highlighted as one of the key technologies for the 5th Generation (5G) and next-generation V2X communication [8, 9]. IEEE 802.11ad, also known as WiGig, is a technical standard for mmWave wireless communications operating in the 60 GHz frequency band which is assigned as an unlicensed band in countries worldwide. The IEEE 802.11ad standard originally defined a high-speed communication mode supporting up to 6.7 Gbps and was further enhanced to support up to 8 Gbps in IEEE 802.11-2016 [10]. Due to its limited communication range and general

need for line-of-sight conditions, small cell architecture has been studied for mmWave V2X communication [11]. To fully utilize the high-throughput performance and low latency of mmWave communication, the use of mobile edge computing/multiaccess edge computing (MEC) has also been studied [12].

In V2X scenarios for a small cell environment, a vehicle establishes a wireless communication link and performs data transmission each time it enters the communication area. To ensure sufficient time for data transmission before the vehicle leaves the communication range, the establishment of the link should be completed as quickly as possible. With the conventional initial link setup process, link establishment often takes several seconds or longer. The fast initial link setup (FILS) was standardized in IEEE 802.11ai [13] and is a method for shortening initial link setup time for sub-6 GHz wireless LAN. FILS achieves a faster link setup by reducing the number of required packet exchanges. A study [14] has proposed introducing the techniques of FILS also for 60 GHz band communication based on IEEE 802.11ad. However, further study may be needed for small cell V2X scenarios because the initial link setup procedure would often be performed at the cell edge while the connecting vehicle is moving, usually resulting in the communication environment fluctuating severely. To benefit from FILS and achieve the fast initial link setup for mmWave V2X, the procedure must be able to handle increased packet loss without incurring substantial delays.

Another challenge for mmWave V2X is the reduced bandwidth due to core network delay and congestion and also cloud server response delay, which prevents full utilization of the high-throughput performance of mmWave communications. MEC is a promising technology to realize applications that require very low latency [15] but is also able to accelerate and improve the efficiency of uploading a large amount of data by using buffering at MEC/mobile edge servers. There has been research on cache size and handover between MEC servers [16, 17]; however, the application programming interface (API) suitable for high-volume data upload using MEC for multi-Gbps wireless access has yet to be presented.

In [18], the authors presented a data collection platform that enables data upload from vehicles to the cloud using IEEE 802.11ad and tackled the challenges mentioned above. This paper describes in detail the system architecture and technologies applied in the data collection platform as an extension of [18] and additionally showcases a proof-of-concept demonstration conducted in Marysville, Ohio, USA. The major contributions of this paper are listed below:

- (1) Presents the proposed fast initial V2X link setup method suitable for IEEE 802.11ad that avoids large delays and extends communication time for a vehicle passing through a communication area (“pass-through communication”)
- (2) Presents REST (Representational State Transfer) API which uses buffering by reverse proxy in the MEC

server for realizing high-speed data upload for multi-Gbps wireless access

- (3) Presents a prototype platform and field evaluations that successfully demonstrate the effectiveness of the proposed methods by showing up to 30 times shorter initial link setup delay and 11 times higher average throughput for data upload during pass-through communication

The rest of this paper is organized as follows. Section 2 describes the system architecture of the data collection platform and the challenges of applying IEEE 802.11ad for V2X scenarios. Section 3 describes two proposed methods which address the technical challenges. Section 4 evaluates the performance of the proposed methods in a field test, and Section 5 shows an experimental demonstration of the data collection platform operating in a real environment. Finally, Section 6 concludes the paper.

2. System Architecture and Target Performance

This section presents the design and technical challenges of 60 GHz small cell radio access for a V2X system based on IEEE 802.11ad.

2.1. Small Cell mmWave for Pass-Through Communication. Table 1 shows the comparison of conventional sub-6 GHz and mmWave communications. Sub-6 GHz wireless communication is suitable for wide area coverage deployment, whereas mmWave wireless communication technologies, such as 5G New Radio (NR) over mmWave and IEEE 802.11ad, are excellent for providing high-throughput communication with focused coverage. This study targets the use case of uploading a large amount of delay-tolerant data from the vehicle to the cloud server. A key objective is to fully utilize the high-throughput performance of mmWave within the limited coverage. The authors assume a vehicle-to-infrastructure (V2I) communication scenario with base stations/roadside units (RSUs) installed at locations such as intersections where the vehicles pass through. Vehicles are equipped with onboard units (OBUs) that upload a large amount of stored data to the cloud via the RSU as they pass through the mmWave coverage of the RSU.

As signal propagation in the mmWave frequency band is highly directional, the RSUs and OBUs employ beamforming, which involves dynamically sweeping/steering the antenna beam direction to achieve wider coverage. Even though the beam sweep direction can be two-dimensional, a one-dimensional (vertical or horizontal) beam sweep is commonly implemented to achieve a small form factor and low cost. In this system, which uses one-dimensional beam sweeping, the antennas of the RSU and the OBU are arranged such that the beams are swept vertically. Figure 1 illustrates the beamforming configuration of the OBU and the RSU. This configuration provides improved coverage in the area below the RSU during V2I pass-through communication.

TABLE 1: Features of wireless communications.

	Sub-6 GHz (4G LTE, 5G NR, Wi-Fi)	mmWave (5G NR, 802.11ad)
Communication area, coverage	Wide coverage (whole area)	Focused coverage (small cell/limited)
Effective throughput performance in a typical usage scenario	Medium (<400 Mbps)	High (>1 Gbps)

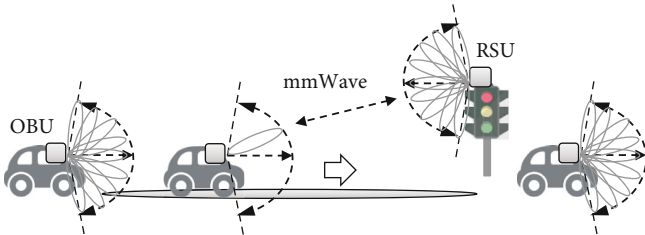


FIGURE 1: Beamforming configuration for V2I pass-through communication. Cited from [18].

2.2. Data Collection Platform. Figure 2 shows the system configuration of the data collection platform. While driving, the vehicle stores data, such as pictures and sensor data, in the OBU. The OBU is equipped with storage to temporarily store the data to be uploaded. The stored data is uploaded when the vehicle enters the RSU's mmWave coverage. The data is transferred to the MEC server via the RSU and from the MEC server to the cloud. The collected data is utilized for visualizing the city. During communication, the OBU takes the role of an IEEE 802.11ad station (STA) and the RSU operates as an access point (AP). The RSU also acts as a router and is connected to the MEC server by 10 GbE; therefore, the OBU gains high-speed access to the MEC server when it enters the RSU's coverage.

2.3. Target Performance. Table 2 shows the target performance of the platform. Here, it is assumed that 1 GB of data is uploaded every hour, which is based on a 1-hour update frequency for quasistatic information of a dynamic map and the approximate amount of data accumulated in a trial vehicle within 1 hour. Supposing a vehicle speed of 60 km/h and a 200m communication area, which is based on throughput measurements conducted in a stationary state [19], the estimated duration available for communication by a vehicle during pass-through communication is 12 seconds. The 12 seconds of communication time can be broken down into 4 seconds for the delay due to the initial link setup and 8 seconds for the data transmission, which results in the target effective throughput of 1 Gbps at the application layer.

2.4. Technical Challenges. In order to achieve the target performance, two major technical challenges need to be addressed. First, IEEE 802.11ad is a wireless standard that assumes a quasistatic environment and the mobility of a pedestrian, so it can take several or more seconds to establish an initial link. This results in insufficient time for data transmission and hence failure to upload the target transfer data amount within a pass-through of the coverage area. Second, the MEC server has access to the Internet to transfer the uploaded data from the OBU to the cloud server, but the

speed of the link between the MEC server and the cloud is much slower than that of the mmWave wireless link; thus, the system would not be able to fully exploit the high-throughput capability of IEEE 802.11ad. The following section addresses these two challenges.

3. Proposed Methods for mmWave V2X

In this section, a fast initial V2X link setup method and high-speed upload using the MEC server are proposed. The fast initial V2X link setup enables the vehicle/OBU to start data transmission shortly after entering the RSU's coverage, thereby increasing data transmission time during pass-through communication. The high-speed upload using the MEC server accelerates data upload by overcoming the limitation of the slower link speed between the MEC server and the cloud.

3.1. Fast Initial V2X Link Setup. This subsection describes the details of the proposed fast initial V2X link setup. The symbols used for the proposed method are given in Table 3.

3.1.1. Initial Link Breakdown Analysis. The factors that take up time in establishing an initial link have been analyzed. As illustrated in Figure 3, the initial link setup procedure consists of three sections, and Figure 4 shows an example sequence of the initial link setup procedure. The breakdown of each section is given below.

(1) PHY (Physical)/MAC (Medium Access Control) Section. The PHY/MAC delay, $T_{\text{PHY/MAC}}$, is the delay from when the first beacon is received from the RSU until the OBU establishes IEEE 802.11ad connection with the RSU. At first, scan processing is performed to search for candidate access points (RSUs) over the specified channels. Then, association processing is performed to connect to the desired access point. The section also includes Wi-Fi Pre-Shared Key (WPA-PSK) authentication processing for encrypted communication and other software processing. Therefore, the PHY/MAC delay, $T_{\text{PHY/MAC}}$, is given as

$$T_{\text{PHY/MAC}} = T_{\text{Scan}} + T_{\text{Assoc}} + T_{\text{WPA-PSK}} + T_{\text{ProcessDelay}} \quad (1)$$

Scan processing is an active scanning procedure defined in IEEE 802.11ad, in which the OBU performs beamforming training by a sector-level sweep sequence upon reception of DMG Beacon frames transmitted by the RSU, followed by a probe exchange. The scan delay, T_{Scan} , is defined as the duration starting from the reception of the first beacon until the end of the active scanning procedure including any scan retries. A scan retry process occurs when the active scanning

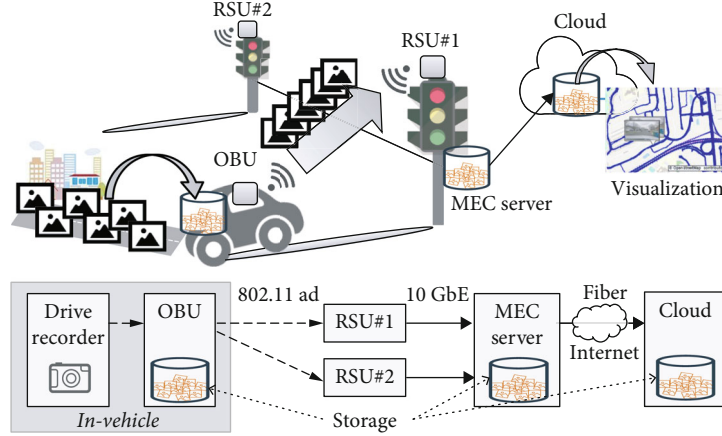


FIGURE 2: System configuration. Cited from [18].

TABLE 2: Target performance of the platform.

	Item	Value
Assumed conditions	Transfer data amount	1 GB
	Vehicle speed	60 km/h
	Communication area	200 m
	Communication time	12 s
Target values	Link setup delay	4 s
	Effective throughput	1 Gbps

time, $T_{ScanTime}$, expires before the OBU successfully completes the probe exchange with the RSU, which results in additional delay. The scan delay, T_{Scan} , is derived as

$$T_{Scan} = (T_{ScanTime} - T_{FirstBeacon}) + (T_{ScanTime} + T_{ScanInterval}) \times N_{ScanRetry}. \quad (2)$$

$T_{ScanTime}$ and $T_{ScanInterval}$ are constants, while $T_{FirstBeacon}$ and $N_{ScanRetry}$ are variables that depend on the channel condition. Association processing consists of join and association frame exchange processes. Join is a process for the OBU to synchronize with the RSU by receiving DMG Beacon frame(s) from the RSU. After a successful join process, the OBU transmits an Association Request frame as part of the association exchange process. If the OBU successfully receives an Association Response frame back from the RSU, the process is completed. Otherwise, the OBU may retry join and association exchange processes after a given backoff time, $T_{AssocBackOff}(n)$. Therefore, the association delay T_{Assoc} is derived as

$$T_{Assoc} = \sum_{n=0}^{N_{AssocRetry}} \{T_{AssocBackOff}(n) + T_{Join}(n) + T_{AssocExchange}(n)\}. \quad (3)$$

The number of retry attempts, $N_{AssocRetry}$, depends on the channel condition. A poor channel condition may result in packet reception errors during join and association exchange processes. $T_{Join}(n)$ depends on the beacon interval, which is typically set to 100 time units (TUs), where 1 TU is 1.024 ms, and the number of beacon intervals until the OBU successfully receives the DMG Beacon frame(s). $T_{AssocExchange}(n)$ may increase if the channel is busy due to communication between the RSU and the other OBUs. There is no backoff time for the first attempt; i.e., $T_{AssocBackOff}(0)$ is 0. For retry attempts, backoff time, $T_{AssocBackOff}(n)$, may be set depending on the current retry count, n . This is discussed in Section 3.1.3 in detail.

(2) *DHCP (Dynamic Host Configuration Protocol) Section.* After establishing the IEEE 802.11ad connection, DHCP processing is performed to assign an Internet Protocol (IP) address to the OBU connected to the RSU. The process includes the negotiation with a DHCP server, duplicate check of the IP address (ping check) by the server, and software processing. The DHCP delay, T_{DHCP} , is derived as

$$T_{DHCP} = T_{dhcp} + T_{PingCheck} + T_{ProcessDelay}. \quad (4)$$

(3) *HTTPS (Hypertext Transfer Protocol Secure) Section.* After establishing the IP connection to the RSU, the uploader application software is immediately launched. Handshake procedures are then performed to establish Transmission Control Protocol (TCP) and Secure Sockets Layer (SSL)/Transport Layer Security (TLS) sessions between the OBU and the MEC server. After all sessions have been established, the data is transferred over HTTPS from the OBU, and an upload completion response is received from the MEC server. Since the transfer time of application data increases according to the amount of data, the delay up to the first byte will be considered the delay of the initial link setup. The section also includes Address Resolution Protocol (ARP) processing to obtain

TABLE 3: Symbols used in the proposed method.

Symbol	Description
$T_{\text{PHY/MAC}}$	Delay from receiving the first beacon to establishing the IEEE 802.11ad connection with the RSU
T_{Scan}	Scan delay
T_{Assoc}	Association delay
$T_{\text{WPA-PSK}}$	WPA-PSK 4-way handshake delay
$T_{\text{ProcessDelay}}$	Processing delay
T_{ScanTime}	Active scanning time
$T_{\text{FirstBeacon}}$	Delay until the first beacon received
$T_{\text{ScanInterval}}$	Scan interval
$N_{\text{ScanRetry}}$	Number of scan retries (probe exchange failures)
T_{Join}	Join delay
$T_{\text{AssocExchange}}$	Channel access time for association frame exchange
$N_{\text{AssocRetry}}$	Number of association retries (association exchange failures)
$T_{\text{AssocBackOff}}$	Association backoff delay
T_{DHCP}	Delay in assigning an IP address to the OBU connected to the RSU
T_{dhcp}	DHCP 4-way handshake delay
$T_{\text{PingCheck}}$	Ping check delay
T_{HTTPS}	Delay from launching the uploader application software to the first byte transfer by HTTPS
T_{ARP}	ARP delay
T_{TCP}	TCP 3-way handshake delay
$T_{\text{SSL/TLS}}$	SSL/TLS 4-way handshake delay
$T_{\text{FirstByte}}$	First byte data transfer delay by HTTPS
T_{Delay}	Delay of the entire initial link setup procedure

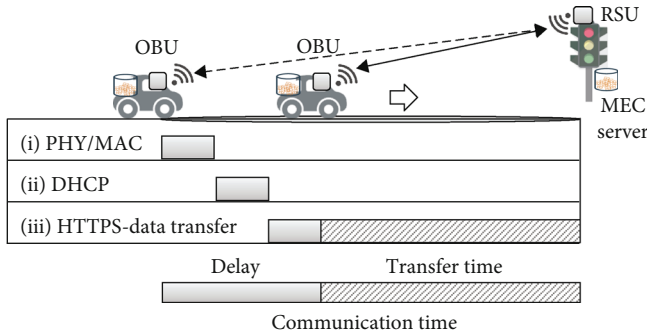


FIGURE 3: Time budget for V2I pass-through communication. Cited from [18].

the MAC address of the RSU and software processing. The HTTPS delay, T_{HTTPS} , is derived as

$$T_{\text{HTTPS}} = T_{\text{ARP}} + T_{\text{TCP}} + T_{\text{SSL/TLS}} + T_{\text{ProcessDelay}} + T_{\text{FirstByte}}. \quad (5)$$

In summary, the delay of the entire initial link setup procedure, T_{Delay} , is given as

$$T_{\text{Delay}} = T_{\text{PHY/MAC}} + T_{\text{DHCP}} + T_{\text{HTTPS}}. \quad (6)$$

Figure 5 shows a breakdown of the measured delays for the conventional link setup method, where delays due to (i) PHY/MAC and (ii) DHCP are observed to be dominant.

3.1.2. Fast Scan Processing and DHCP Processing. Conventionally, when scanning for APs, the scan period, which is the active scanning time, T_{ScanTime} , is set to several seconds. To analyze the impact of the scan period, the scan success rates for different scan periods were evaluated. As shown in Figure 6, the scan success rate gradually decreased as the scan period was reduced to 210 TUs and then decreased sharply when below 210 TUs. The sharp decrease in the success rate was due to the OBU not receiving a beacon at least once during the scan period. This occurred as the scan period became shorter than the beacon interval, which was set to 100 TUs, plus the operating speed of the firmware (time to start scanning the specified channel). Therefore, a scan period of 210 TUs ($=T_{\text{ScanTime}}$) was selected to minimize delay while ensuring an acceptable scan success rate. Additionally, the probe exchange performed after reception of the first beacon could fail due to packet errors depending on the channel condition. Especially if there are multiple vehicles/OBUs, collisions and carrier sense delays may occur

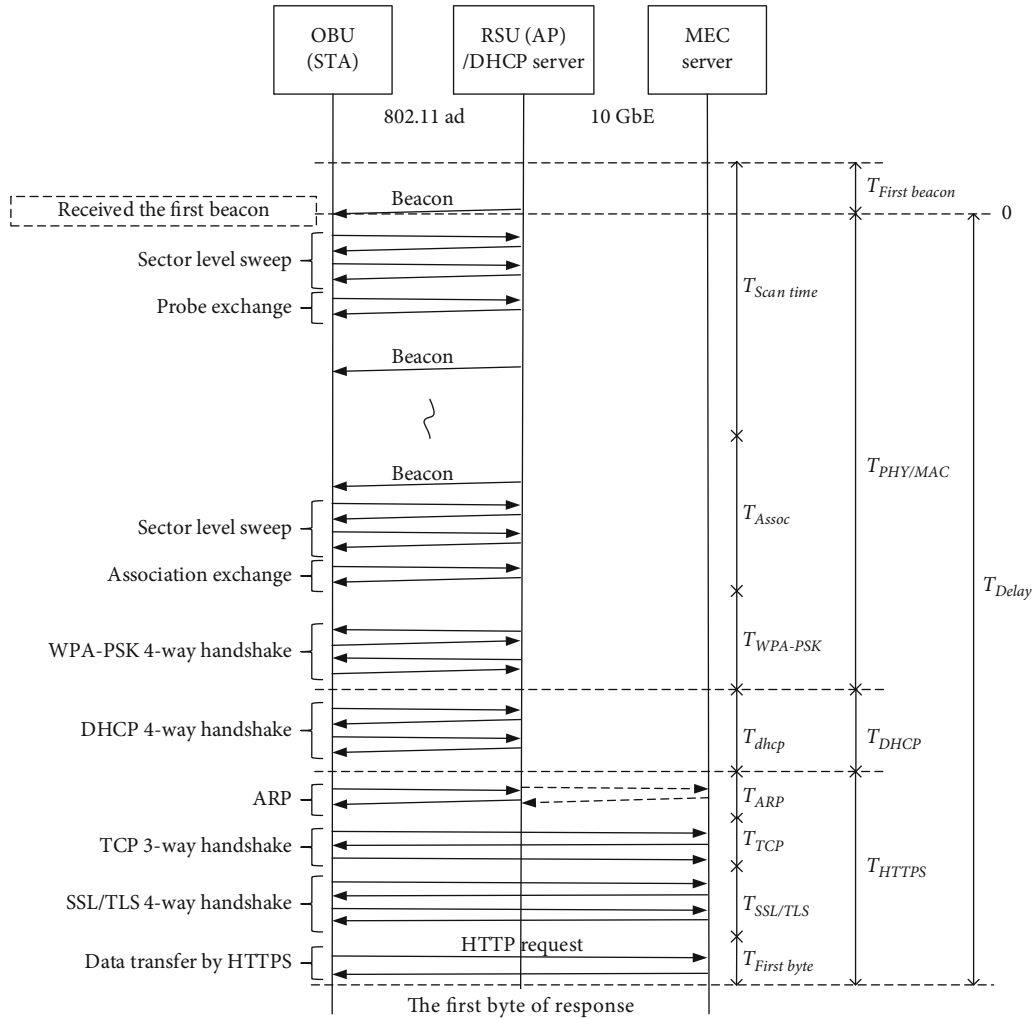


FIGURE 4: Initial link setup procedure example sequence with analysis.

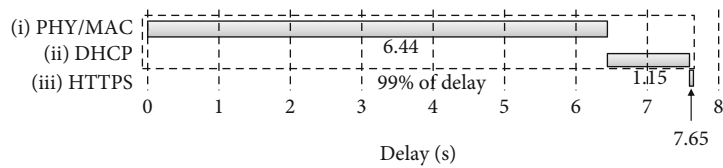


FIGURE 5: Delay performance with the conventional method. Cited from [18].

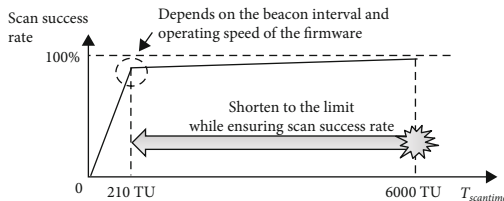


FIGURE 6: Relationship between the scan period and the scan success rate. Cited from [18].

due to the IEEE 802.11ad Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA). Therefore, to avoid such delays, $T_{ScanInterval}$ was set to 0 to make the OBU immediately retry scan processing (rescan) after an unsuccessful

scan attempt. By these configurations, the scan period was shortened by up to about 6 seconds.

Typically, the DHCP server needs to perform a duplicate check to confirm that no other OBU is using the same address to be offered before notifying the OBU of the IP address information. But in this system, the DHCP server is installed in the RSU, and the IP address is assigned only to the terminal connected to the RSU. Therefore, the duplicate check may be skipped; i.e., set $T_{PingCheck}$ to 0. By doing so, the DHCP time was shortened by about 1 second compared to the conventional implementation. Table 4 summarizes the values that were applied for configurable parameters related to PHY/MAC and DHCP processes.

TABLE 4: Parameters for fast scan and DHCP processing.

Section	Item	Value	
		Conventional method	Proposed method
$T_{PHY/MAC}$	$T_{ScanTime}$	6000 TUs	210 TUs
	$T_{ScanInterval}$	5 s	0 s
T_{DHCP}	$T_{PingCheck}$	1 s	0 s

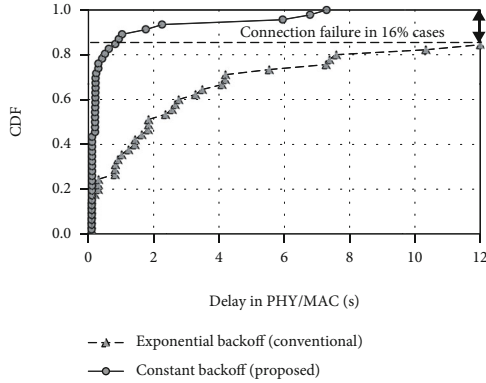


FIGURE 7: Comparison of PHY/MAC delay performance in a mobility environment. Cited from [18].

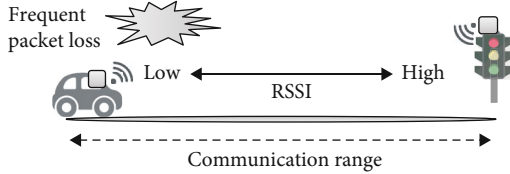


FIGURE 8: Wireless environment in V2I communication where frequent packet loss occurs. Cited from [18].

3.1.3. *Reduction of Delay Variation due to Association Processing Failure.* Figure 7 shows the cumulative distribution function (CDF) of the PHY/MAC delay in a mobility environment (at 60 km/h) after applying the improvements described in Section 3.1.2. The PHY/MAC delay in the best case has been reduced to well below 1 second as a result of applying the techniques on scan processing and DHCP described in Section 3.1.2. It can also be observed from the conventional method that large delays exceeding 8 seconds occur from 80% in the CDF, and connection failure, in which the OBU could not complete the establishment of the connection within the target 12 seconds, occurred in 5 out of the 45 test runs (16%). The large variations in delay were determined to be from the association process. In the WPA supplicant software [20], which controls the association process, exponential backoff ($T_{AssocBackOff}(n)$) is applied upon failure of the Association Request/Response frame exchange. The backoff durations configured in the WPA supplicant software for the first to fourth failures are 100, 500, 1000, and 5000 ms, respectively.

In the pass-through communication scenario, the association process is expected to be performed at the edge of the

TABLE 5: Parameters for the association of a mobility environment.

Section	Item	Value	
		Conventional method	Proposed method
$T_{PHY/MAC}$	$T_{AssocTimeOut}$	10 s	0.24 s
	$T_{AssocBackOff}(n)$	Exponential backoff 0.1, 0.5, 1, 5 s for $n = 1, 2, 3, 4$	0.05 s for any $n \geq 1$

TABLE 6: An example of a POST request for hundreds of pictures.

```
Endpoint: /edgeapi/v1/storage/file/_multipart
Request headers:
Content-Type: multipart/form-data; boundary = boundary
Content-Length: Length

Request body:
-boundary
Content-Disposition: form-data; name = "metadata"
Content-Type: application/json; charset = UTF-8

<list of metadata in JSON format>
-boundary
Content-Disposition: form-data; name = "jpgfile"; filename = "000001.jpg"
Content-Type: image/jpeg

<binary content of file: 000001.jpg>
-boundary
...
-boundary
Content-Disposition: form-data; name = "jpgfile"; filename = "000100.jpg"
Content-Type: image/jpeg

<binary content of file: 000100.jpg>
-boundary
```

TABLE 7: PC specs for performance measurement.

Component	Spec
CPU	Intel Core i7-6600U, 2.6 GHz
RAM	DDR4 2133 MHz, 16 GB
SSD	SATA III, 500 GB (OBU and RSU)/1 TB (MEC server)
OS	Ubuntu 16.04

mmWave coverage as the OBU enters the RSU’s communication range. As a result, packet loss for the Association Request/Response may occur frequently. This is illustrated in Figure 8. Thus, the WPA supplicant software was modified to apply a constant backoff time of 50 ms after the failure of the association exchange and the timeout of the association process was set to 240 ms so that the OBU retries the association exchange up to 4 times. In case of a timeout, the OBU restarts with scan and join processes to perform

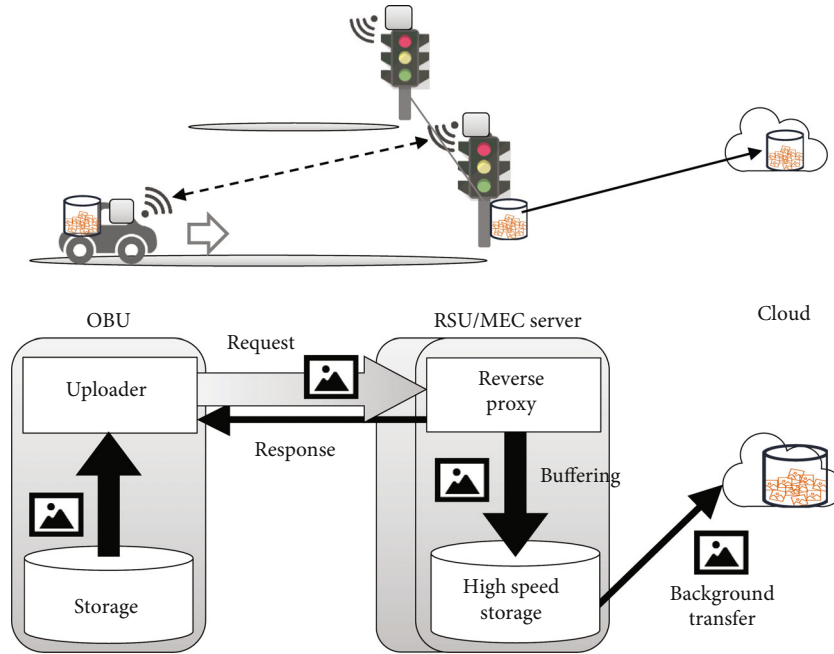


FIGURE 9: System configuration/protocol using the MEC server. Cited from [18].

synchronization with the RSU. This avoids harmful interference from the continuous transmission of Association Request frames by the OBU when the RSU is busy or unavailable. Table 5 summarizes the parameters for conventional and proposed methods. As shown by the solid line in Figure 7, the proposed method was able to substantially suppress the variations in delay.

3.2. High-Speed Upload System Using the MEC Server. This subsection describes the details of the proposed method of high-speed upload using the MEC server including overcoming the issue of slower link speed between the MEC server and the Internet.

3.2.1. High-Speed Upload API Design. In this subsection, a design of REST API that is suitable for high-speed upload during pass-through communication is presented. REST API is usually HTTP-based and implements CRUD (Create, Read, Update, and Delete) operations on resources on the remote server. Recently, HTTPS-based REST API is becoming widely adopted for Internet applications. REST API is also being considered to be adopted for MEC applications, and [15] describes design principles for REST MEC service APIs. While most typical applications for MEC involve offloading of tasks that require large computational cost and/or low latency which cannot be fulfilled by on-vehicle computers, upload acceleration may also be considered a class of task offloading. With REST API, the upload task is implemented as an HTTP POST request.

In the data collection platform, thousands of photo images are uploaded by a vehicle during pass-through communication. While upload time for an image, which is several hundreds of KBs, takes only several milliseconds, the

response time of the server may take at least several milliseconds for each request, which causes degradation of throughput performance, especially if each image is uploaded with a single POST request. Bundling is a technique that is used to reduce protocol overhead by uploading a batch of files [21]. Bundling also reduces performance requirements for the PCs of both the server and the OBU since it reduces the number of HTTP requests to be processed. As shown in Table 6, the proposed API applies the multipart/form-data format that is defined in RFC 7578 [22] to bundle hundreds of image data in a single POST request. A set of metadata attributed to the images to be uploaded, such as capture location data based on GNSS and vehicle ID, is bundled to the same request as well.

At the OBU, from the total stored image data of about 10 GB, around 100 MB of image data, which consists of almost 200 image files, is bundled in a single POST request by the upload client software, and up to two requests are issued concurrently in two worker threads to achieve a sustained data rate of over 1 Gbps. The 100 MB size was selected to balance the trade-off between higher transfer efficiency and loss of performance due to failure of an ongoing upload due to connection loss. The throughput performance of the data upload using the proposed HTTPS-based REST API could reach more than 2.8 Gbps in an indoor environment even with the use of industrial PCs with modest specifications (as shown in Table 7) for the server and the OBU.

3.2.2. Upload Acceleration Using the Reverse Proxy-Based MEC Server. With the proposed REST API and OBU implementation described in the previous subsection, high-speed upload to the MEC server over mmWave communication is achieved. However, the bandwidth and delay of the



FIGURE 10: Field test environment in Singapore. Cited from [18].

Internet line between the MEC server and the cloud becomes a bottleneck. As illustrated in Figure 9, the MEC server is equipped with a reverse proxy function that terminates HTTPS access from the OBU and buffers the uploaded data from the OBU. As a result, high-capacity transmission between the OBU and the MEC server can be realized by the MEC server which functions as a proxy, returning a high-speed response to the OBU directly without being limited by slower Internet communication speeds between the MEC server and the cloud. The server then transfers the buffered data to the cloud server as a background process even after the vehicle has passed through.

4. Measurement Results

This section shows the measurement results of a field test. The purpose of the field test was to measure the performance of high-capacity data upload during pass-through communication by a single vehicle with 60 km/h speed. The effectiveness of the technologies proposed in Section 3 is discussed based on the measurement results.

4.1. Field Test Environment. Figure 10 shows the field test environment which is a public road in Singapore. One RSU and one MEC server were installed at a bus stop as an experimental setup, and an OBU was installed inside a test vehicle. The test was conducted with the developed data collection platform that implemented the proposed methods described in Section 3, including the countermeasures described in Sections 3.1 and 3.2. The OBU on the moving vehicle performed pass-through communication to upload image data to the bus stop setup. Table 8 shows the experimental configuration, and the PC specifications for the RSU, OBU, and MEC servers are shown in Table 7 in Section 3.2.1.

4.2. Test Results. Figure 11 shows the CDF of the initial link setup delay performance. When employing the proposed method, the shortest delay was 0.25 seconds, which is a 30 times reduction from the conventional method's 7.65 sec-

TABLE 8: Experimental configuration.

Item	Value
Communication module	Peraso [23] X710
Wireless protocol	IEEE 802.11ad
Carrier frequency	60.48 GHz
Modulation and coding scheme (MCS)	0~12
Beam sweep direction	Vertical
Antenna polarization	Vertical
RSU antenna height	3.0 m
OBU antenna height	1.2 m (inside vehicle)
Number of vehicles	1
Number of OBU, RSU, and MEC servers	1 unit each
Vehicle speed	About 60 km/h
Throughput measurement method	Measured on HTTPS
Number of test runs	46

onds shown in Figure 5. The establishment of the initial link was successfully completed within the communication area for all 46 test runs, and in about 94% of cases (=43 test runs), the measured delay was better than the target delay of 4 seconds. The reduction of delay and suppression of delay fluctuations after association processing failure in pass-through communication were achieved. Still, about 6% of cases, as shown in Figure 11, took 4 seconds or more. This was due to the PHY/MAC wireless environment, as evident from the solid line's behavior in Figure 7. It is suspected that these occurred when no beacons were received for some time after an initial beacon reception at a far distance and will be investigated further in the future.

Figure 12 shows the complementary cumulative distribution function (CCDF) of average throughput performance during pass-through communication for uploading image data. By employing the proposed methods, the median value of the entire 46 test runs was 1.1 Gbps, which is 11 times higher than the conventional 0.1 Gbps. In addition, the

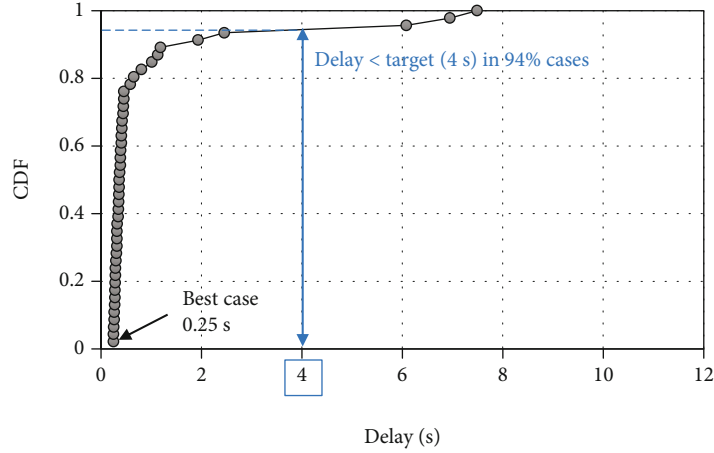


FIGURE 11: Delay performance with the proposed method. Cited from [18].

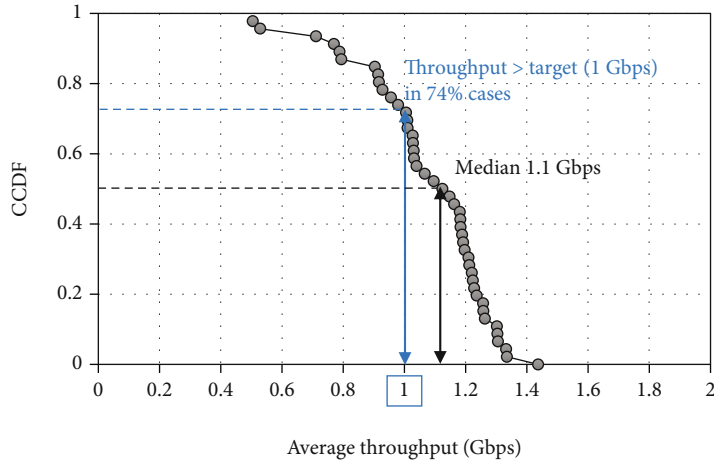


FIGURE 12: Average throughput performance with the proposed method. Cited from [18].

target of 1 Gbps or more was achieved in about 74% of cases (=34 out of 46 test runs). Although the measured throughput satisfied the target performance, it was still lower than expected. The authors analyzed and improved on the data upload client software in the OBU as described in Section 5.2 after this test.

Tables 9 and 10 show the benchmark comparisons that summarize the above results. In summary, the delay for the initial link setup was shortened by 30 times (7.65 seconds to 0.25 seconds), and about 94% of cases achieved the target of 4 seconds or less. In addition, the average throughput for uploading image data was improved by 11 times (0.1 Gbps to 1.1 Gbps), and the target of 1 Gbps or more was achieved in about 74% of cases. These results show that it is possible to consistently upload a large amount of data from a moving vehicle to the MEC server on the roadside, which was challenging in the past based on the conventional methods.

5. Proof of Concept

This section presents a demonstration of the data collection platform incorporating data visualization application software, which was trialed in the city of Marysville, Ohio, USA.

TABLE 9: Performance comparison.

	Delay	Throughput
Conventional	7.65 s	0.1 Gbps
Proposed	0.25 s	1.1 Gbps
Improvement rate	30 times	11 times

TABLE 10: Achievement rate of the target.

	Delay	Throughput
Conventional	$\ll 1\%$	0%
Proposed	94%	74%

5.1. System Installation. The prototype platform was installed, as shown in Figure 13, in early 2020 with the cooperation of Marysville City Hall, which is leading advanced initiatives for smart cities. RSUs and the MEC server were installed at the intersection of 6th/Main Street, Marysville, and the OBU system including a drive recorder, which consisted of a smartphone with a camera, was installed on a utility vehicle operated by the city. The system on the vehicle

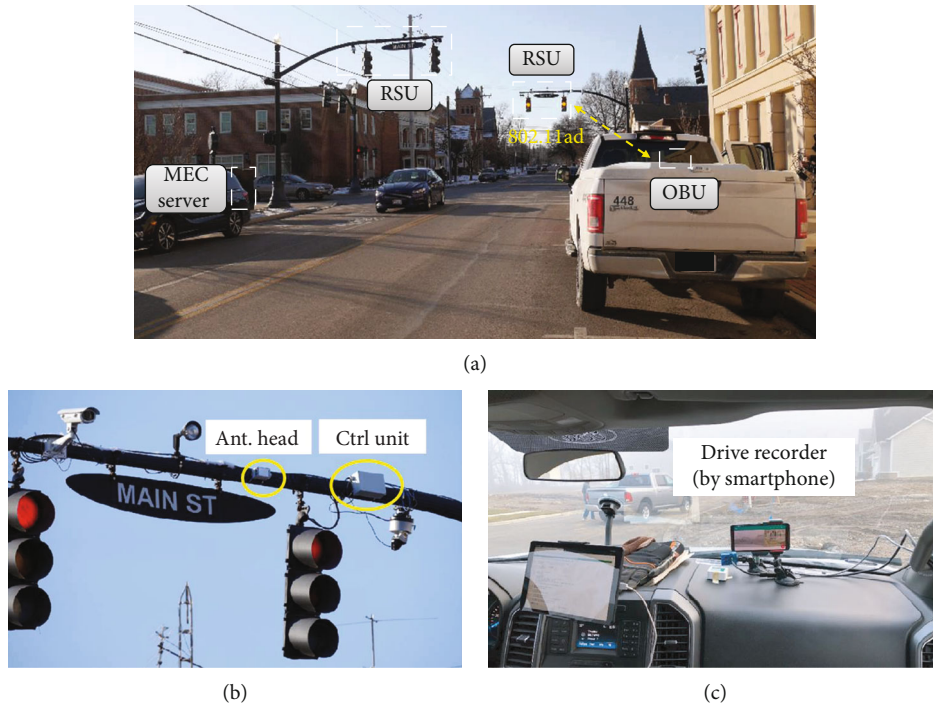


FIGURE 13: Experimental demonstration environment in (a) 6th/Main Street, Marysville. (b) RSU and (c) OBU installation setup.

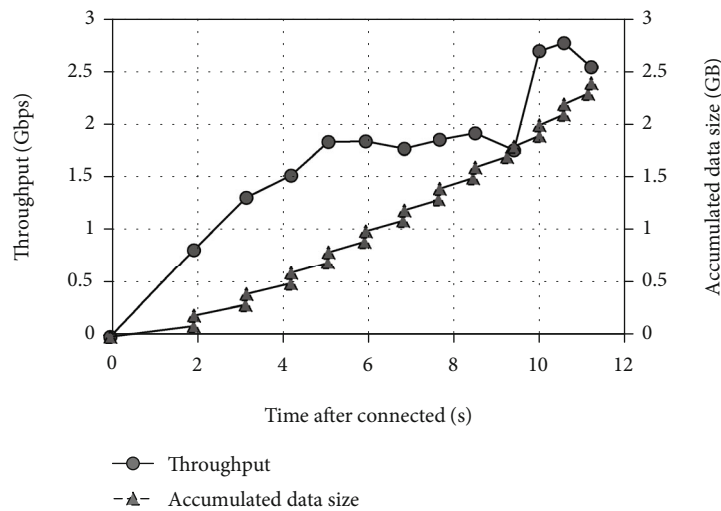


FIGURE 14: Throughput and accumulated data size performance in the trial.

automatically collected images of the city as the vehicle was used for operations by the city staff. In the prototype system, the vehicle’s drive recorder takes a picture every second and continuously transfers the image data to the OBU via Wi-Fi (2.4 GHz/5 GHz) and stores it in the OBU’s storage. When the vehicle approaches the intersection, the system automatically uploads the collected image data, which is then processed by the application software for visualizing the city.

5.2. *Demonstration.* Figure 14 shows an example of the measured performance during pass-through communication for

uploading image data. The throughput is indicated by the solid line and the left axis, which shows a gradual increase after connection and a peak throughput of 2.8 Gbps at the application layer. The antennas of the RSU and the OBU were arranged so that the beam is swept vertically as described in Section 2.1, resulting in a high throughput just before the disconnection of V2I pass-through communication. The accumulated data size uploaded in two worker threads is indicated by the dashed line and the right axis, and it shows that a total amount of 2.4 GB of data was uploaded within the 11.2 seconds of pass-through communication at the intersection.

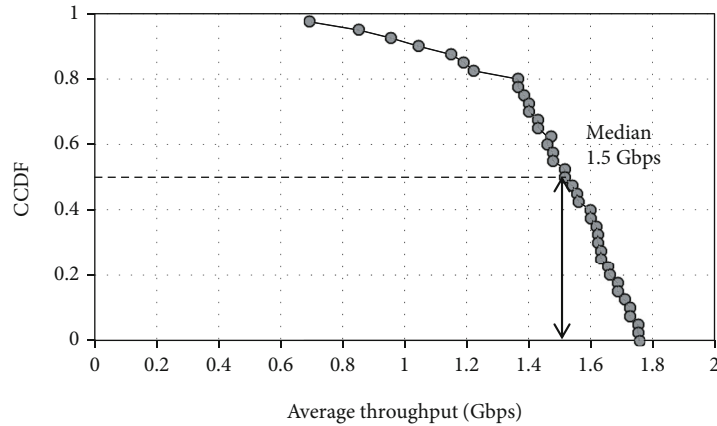


FIGURE 15: Average throughput performance in the trial.

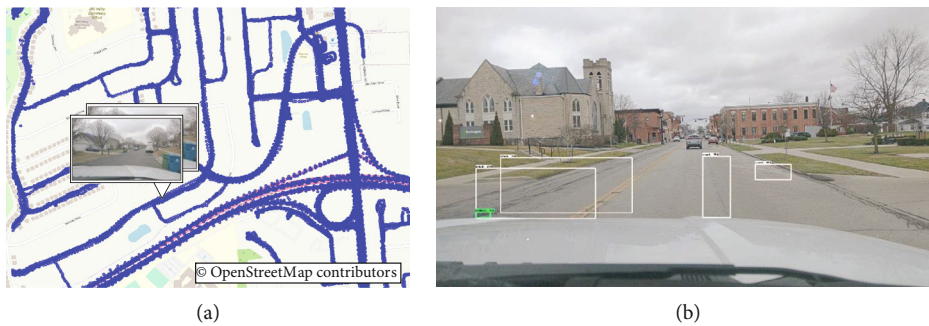


FIGURE 16: City visualization using collected data in a demonstration. (a) Dashboard and (b) AI-analyzed image with detected road defects.

Figure 15 shows the CCDF of average throughput performance measured during the trial. The median value of the entire 40 runs was 1.5 Gbps, and higher throughput was observed compared to the field test in Singapore. This improvement may be attributed to optimizations in the data upload client software at the OBU to reduce unnecessary memory copies and simplify file selection for bundled data upload. In addition, due to the regulatory rules for the 60 GHz band in the US allowing higher transmit power than in Singapore, the range of communication for 16 QAM, which realized over 2 Gbps throughput, was increased. Slower vehicle speed during the US test due to the actual traffic condition in the city may have also contributed to the higher average throughput.

During the trial, image data of 10 GB or more per vehicle was collected daily by the cloud, and the data could be searched and browsed. Figure 16(a) shows a dashboard that visualizes the city using the collected data. Blue dots correspond to the image data taken by the camera on the vehicle. As an example, the collected image data were used for detecting road defects. As shown in Figure 16(b), the post-processing analytics using artificial intelligence (AI) successfully detected dents and cracks on the road. By detecting the road defects in their early stage, the city can take proactive action for repair, leading to reduced personnel and maintenance costs. There are tremendous opportunities for using

the data to derive insights and take actions for the benefit of the city. This work paves the way for improving the quality of services in smart cities.

6. Conclusions

This paper has presented a data collection platform that realizes high-capacity data upload from vehicles to the cloud using the next-generation gigabit V2X communication using 60 GHz mmWave based on IEEE 802.11ad. The platform features the proposed fast initial V2X link setup and high-capacity data upload using a MEC server, which enables full utilization of the high-throughput performance of mmWave communication. The evaluation results show a 30 times shorter delay and 11 times higher average throughput from the vehicle to the cloud in a typical V2X pass-through scenario. As a proof of concept, the platform was installed in the city of Marysville and successfully demonstrates the potential for solving problems in smart cities through high-capacity data collection, analytics, and city visualization. Further work is planned for evaluating the initial link setup according to the distance between the vehicle/OBU and the RSU based on GNSS position information and also for evaluating the simultaneous connection/communication from multiple vehicles.

Data Availability

The data generated or analyzed during this study are included within the article.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

The authors would like to thank Ryo Takahashi, Fumihiko Onoda, and Kyosuke Mishima with Panasonic Automotive Systems Company of America; Mike Andrako, Darick Moore, Mark Dilsaver, and Terry Emery with the city of Marysville, Ohio; and Eric Philips with Union County, Ohio, for their invaluable support for the demonstration experiment in the city of Marysville, Ohio. Funding was provided by in-house R&D expenses.

References

- [1] A. D. Angelica, "Google's self-driving car gathers nearly 1 GB/sec," March 2022, <http://www.kurzweilai.net/googles-self-driving-car-gathers-nearly-1-gbsec>.
- [2] "Data is the new oil in the future of automated driving," March 2022, <https://newsroom.intel.com/editorials/krzanich-the-future-of-automated-driving/>.
- [3] K. H. Law and J. P. Lynch, "Smart city: technologies and challenges," *IT Professional*, vol. 21, no. 6, pp. 46–51, 2019.
- [4] U. Yutaka, "Start-up of data utilization-type smart cities," *NEC Technical Journal*, vol. 13, no. 1, 2018.
- [5] F. Z. Benhamida, A. Bouabdellah, and Y. Challal, "Using delay tolerant network for the Internet of Things: opportunities and challenges," in *2017 8th International Conference on Information and Communication Systems (ICICS)*, pp. 252–257, Irbid, Jordan, April 2017.
- [6] "IEEE Standard for Information Technology—local and metropolitan area networks—specific requirements—part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 6: wireless access in vehicular environments," in *IEEE Std 802.11p-2010*, July 2010, <https://standards.ieee.org/ieee/802.11p/3953/>.
- [7] "Study on LTE-based V2X services," in *3GPP TR 36.885, vol. 14.0.0*, July 2016, <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2934>.
- [8] K. Sakaguchi, R. Fukatsu, T. Yu et al., "Towards mmWave V2X in 5G and beyond to support automated driving," *IEICE Transactions on Communications*, vol. E104.B, no. 6, pp. 587–603, 2021.
- [9] T. Okuyama, S. Suyama, N. Nonaka, Y. Okumura, and T. Asai, "Outdoor experimental trials of millimeter-wave base station cooperation with digital beamforming in high-mobility environments for 5G evolution," in *2020 IEEE 92nd vehicular technology conference (VTC2020-fall)*, pp. 1–5, Victoria, BC, Canada, Nov. 2020.
- [10] "IEEE Standard for Information Technology—telecommunications and information exchange between systems local and metropolitan area networks—specific requirements - part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications," in *IEEE Std 802.11-2016*, Dec. 2016, <https://standards.ieee.org/ieee/802.11/5536/>.
- [11] K. Sakaguchi, T. Haustein, S. Barbarossa et al., "Where, when, and how mmWave is used in 5G and beyond," *IEICE Transactions on Electronics*, vol. 100, no. 10, pp. 790–808, 2017.
- [12] P. Mach and Z. Becvar, "Mobile edge computing: a survey on architecture and computation offloading," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1628–1656, 2017.
- [13] "Amendment 1: fast initial link setup," in *IEEE Std 802.11ai-2016*, 2016, <https://standards.ieee.org/ieee/802.11ai/5017/>.
- [14] H. Kushida, H. Mano, M. Takai, Z. Liu, and S. Ishihara, "On the effectiveness of FILS in IEEE 802.11ad wireless networks," in *2017 23rd Asia-Pacific conference on communications (APCC)*, pp. 1–6, Perth, WA, Australia, Dec. 2017.
- [15] "Multi-access edge computing (MEC); general principles, patterns and common aspects of MEC service APIs," in *ETSI GS MEC 009 V3.1.1*, June 2021, https://www.etsi.org/deliver/etsi_gs/MEC/001_099/009/03.01.01_60/gs_MEC009v030101p.pdf.
- [16] Z. Zhang, Z. Chen, and B. Xia, "Cache-enabled uplink transmission in wireless small cell networks," in *2018 IEEE International Conference on Communications (ICC)*, pp. 1–6, Kansas City, MO, USA, May 2018.
- [17] S. D. A. Shah, M. A. Gregory, S. Li, and R. D. R. Fontes, "SDN enhanced multi-access edge computing (MEC) for E2E mobility and QoS management," *IEEE Access*, vol. 8, pp. 77459–77469, 2020.
- [18] K. Nakano, A. Egami, H. Motozuka et al., "Data collection platform for smart city with gigabit V2X communication over 60 GHz band," in *2021 IEEE 93rd vehicular technology conference (VTC2021-spring)*, pp. 1–5, Helsinki, Finland, April 2021.
- [19] M. Irie, G. W. Y. Huang, M. S. H. Cheng, and K. Takahashi, "Bulk sensor data sharing using millimeter wave V2X for enhanced safety and comfort in mobility," in *2018 Asia-Pacific microwave conference (APMC)*, pp. 186–188, Kyoto, Japan, Nov. 2018.
- [20] "Linux WPA/WPA2/IEEE 802.1X supplicant," March 2022, https://w1.fi/wpa_supplicant/.
- [21] M. Akter, A. Gani, M. O. Rahman, M. M. Hassan, A. Almogren, and S. Ahmad, "Performance analysis of personal cloud storage services for mobile multimedia health record management," *IEEE Access*, vol. 6, pp. 52625–52638, 2018.
- [22] L. Masinter, "Returning values from forms: multipart/form-data," in *RFC 7578, IETF*, Jul. 2015, <https://datatracker.ietf.org/doc/html/rfc7578>.
- [23] "mmWave 60GHz chipset products," June 2022, <https://perasotech.com/60ghz-hardware/>.