https://doi.org/10.3311/PPtr.18402



Periodica Polytechnica Transportation Engineering, 51(1), pp. 57–69, 2023

# Service-Oriented Architecture for Providing ITS Services in Vehicular Ad Hoc Networks

Kifayat Ullah[1,2*], Ume Habiba[2]

[1] Department of Computer Science, University of São Paulo, 1010 Rua do Matão, 05508-090 São Paulo, Brazil
[2] Department of Computer Science, CECOS University of IT and Emerging Sciences, Street 1, Sector F 5 Phase 6 Hayatabad, 25000 Peshawar, Pakistan
[*] Corresponding author, e-mail: kifayat@cecos.edu.pk

Received: 21 April 2021, Accepted: 26 September 2022, Published online: 06 October 2022

**Abstract**

Vehicular Ad hoc Network (VANET) is the cutting edge technology for smart transportation. VANET becomes an important aspect of the Intelligent Transport System (ITS). Different safety and non-safety applications have been developed for VANET. The inspiration behind VANET is to provide safe, and pleasant journeys to the drivers and passengers. Although the quality of software depends upon its architecture, most of them do not give proper attention to the consideration of Software-Oriented Architecture (SOA) for providing safety and non-safety ITS services in VANET. To address this issue, we proposed an efficient software architecture by highlighting the important operations and services of the system. The performance of the proposed architecture is evaluated by several design metrics and the results are compared with a state-of-the-art solution. The results showed that our proposed architecture has low coupling and high cohesion factors. Furthermore, the results reveal that our architecture is less complex and more reusable. From the results, we conclude that the proposed architecture is suitable for providing safety and non-safety ITS services and will pave the way for the implementation of the futuristic vision of the ITS.

**Keywords**

SOA, VANET, ITS services, smart transportation, pothole detection, design metrics

## 1 Introduction

Smart City (SC) makes use of Information and Communication Technologies (ICT) to enhance the living standard of citizens and improve the quality and performance of urban services such as energy, transportation, and utilities. Some advantages of SC are sustainability, smart living, connectivity, comfort, and efficient resource utilization (Nagy and Csiszár, 2020). It is an emerging concept that attracts the attention of researchers, policy-makers, city administrators, and government agencies, throughout the world. The main focus of SC is to address the problems of the urban population by using the latest sensing, analyzing, and communication technologies and standards.

The Intelligent Transportation System (ITS) is playing a key role in the SC. The driving forces behind ITS make it possible to support the SC vision, which aims at employing advanced and powerful communication technologies for the administration of the city and the citizens. The ITS will also reshape the means of transportation. It would allow passengers to easily select different transportation options for the lowest cost, shortest distance, or fastest route. As an example, ITS provides real-time traffic information to citizens in order to inform them about congested roads and traffic conditions. The success of ITS not only contributes to the SCs but also to smart living and smart environments. It is also a step towards green transportation and green cities.

VANET provides a platform for the deployment of a large number of safety and non-safety applications. These applications ensure the safety of drivers and passengers on roads. It also enhances the traveling experience by allowing access to different services on the way. In addition to the benefits, VANET poses many challenges, such as quality of service, high connectivity demand, bandwidth utilization, delay, security, and privacy. In a typical VANET environment, vehicles communicate with each other through Vehicle to Vehicle (V2V) communication in an ad hoc fashion. Additionally, the vehicles are able to communicate with the roadside infrastructure through Vehicle to

**Cite this article as:** Ullah, K., Habiba, U. (2023) "Service-Oriented Architecture for Providing ITS Services in Vehicular Ad Hoc Networks", Periodica Polytechnica Transportation Engineering, 51(1), pp. 57–69. https://doi.org/10.3311/PPtr.18402

Infrastructure (V2I) communication. In V2I communication, the Road Side Unit (RSU) provides the services of a base station and it works as a fixed infrastructure (Ullah et al., 2019). Even though a great quantity of both safety and non-safety applications are proposed for VANET, most of them do not give proper attention to the use of software architecture. Regardless of the fact that the life and quality of these applications depend upon its architecture.

Software architecture is a fundamental component of a software system. It serves as a blueprint for the design of the software system. Its consideration is also important for the quality design of the system, as desired by the stakeholders. It is the high-level design of a software system for creating and connecting software components concerning different stakeholder views (Khan et al., 2016). The architecture facilitates reasoning and change management during different phases of the complex software life cycle. Due to changes in the nature of software systems, the architectures have also evolved in the past decades (Woods, 2016). In this regard, the SOA is one of the latest models where services are provided to other components through a communication protocol over a network. The main advantage of SOA is that it is independent of different vendors, products, and technologies.

Besides the importance of a SOA, current solutions for VANET give little attention to the utilization of software architectures. As a result, the developed solutions (protocols, models, etc.) are difficult to reproduce and compare. In addition, the problems of the platform and underlying network dependencies, high coupling, re-usability, and implementation complexities are also inherited in the solutions.

To address these problems, we proposed an SOA for providing ITS services in VANET. In our solution, smart vehicles play an important role. We believe that the adoption of our well-structured and robust software architecture would facilitate the implementation of various services and applications. Furthermore, it will help the researchers, software developers, and policy makers to develop better software systems. Our proposed software architecture is depicted in Fig. 1. Section 3 further explains the proposed SOA.

The main contributions of this research work are:

- To design a service oriented architecture that will ease the deployment of both safety and non-safety ITS applications.
- To identify the main services and operations of the proposed system.
- To evaluate the performance of the proposed solution in terms of coupling, cohesion, and complexity.

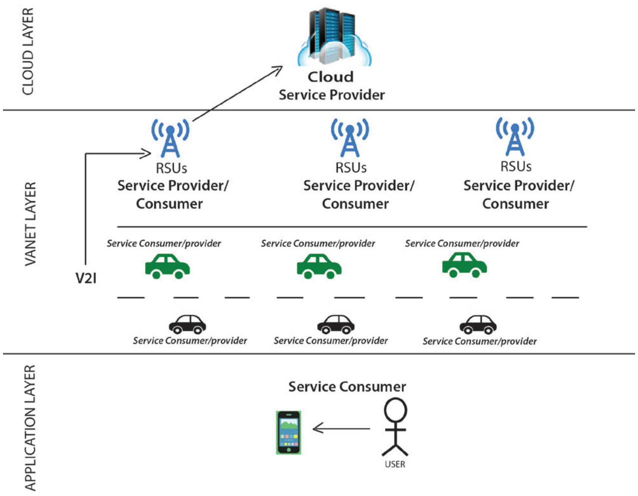


**Fig. 1** Proposed architecture

## 2 Literature review

In the literature different solutions have been proposed for the provision of different ITS applications. An image processing-based solution for addressing the problem of road health monitoring is proposed in (Moazzam et al., 2013). The work presents a low-cost model for analyzing 3D pavement distress images. The proposed solution makes use of a low-cost Kinect sensor that gives direct depth measurements, thereby reducing computation costs. The Kinect sensor consists of RGB and infrared cameras. These cameras capture RGB and depth images. For analysis of the collected images, a MATLAB environment was used.

A model to detect the three-dimensional cross-section of pavement potholes is proposed in (Youquan et al., 2011). The proposed method makes use of Light Emitting Diode (LED) linear light and two Charge Coupled Device (CCD) cameras to capture pavement images. The proposed method also employs various digital image processing techniques, e.g., binarization, thinning, three-dimensional reconstruction, error analysis, and compensation to get the depth of potholes. The main drawback of this method is that the results get affected by LED light intensity and other environmental factors.

A Support Vector Machine (SVM) based solution for pothole detection is proposed in (Lin and Liu, 2010). One of the advantages of the SVM based method is that it distinguishes potholes from cracks, and bumps etc. The proposed method segments the images by using partial differential equations. To detect potholes, the method trains the SVM with a set of pavement images. However, the training model fails, if the images are not properly illuminated.

A vibration-based pothole detection technique is proposed in (Strutu et al., 2013). The proposed method uses an accelerometer sensor, and GPS device to identify the precise location of potholes. The mobile node (i.e., moving vehicle) is equipped with a wireless router, an accelerometer sensor, a GPS device, and computer. The computer runs the pothole detection algorithm. The collected data is transmitted to a central location for processing and analysis. The main disadvantage of this technique is the dependency on the wireless router and computer, mounted in the vehicle.

Another solution to address the problem of roadside service provision is proposed in (Singh et al., 2018). This work provides a discussion on pothole detection systems. In addition, it also proposed a solution to recognize bumps, potholes, and speed breakers on the road surface and give up-to-date notifications to the drivers. For this purpose, ultrasonic sensors and GPS devices are used. As mobile devices are battery-powered with limited processing capabilities, it would not be possible to use them for continuous monitoring of road surfaces.

The study (Arif and Wang, 2020) presents a cloud-based SOA for the establishment of V2V and V2I communication. The proposed architecture is composed of five layers namely the application layer, application service layer, services layer, cloud infrastructure layer, and a layer of vehicles. Although V2V, V2I communication, and SOA are essential for providing different services, no proper attention is given to the software design components and services (e.g., service composition, operation, and interconnection).

An important service of a smart city is the parking management system. An integrated parking management system is presented in (Sándor and Csiszár, 2015). The proposed work provides an efficient model for addressing the limitations of the local methods for parking management. The proposed parking management system provides a real-time solution by considering the actual demands and capacity of parking slots, traffic conditions, and control strategies. The authors also provide a detailed overview of the different functions of the proposed information system. Furthermore, they presented both structural architecture and operational models for the parking management system by highlighting the different entities involved in them. The proposed parking system is recommended for traffic management which is helpful for both the end-users and operators. A major limitation of this work is the lack of simulations. The authors did not perform any simulation to implement, test, evaluate, and compare the performance of their proposed solution.

A systematic literature review about the quality of smart mobility is presented in (Nagy and Csiszár, 2020). The main focus is to explore the quality elements in the context of smart cities. Three main areas of quality concepts, i.e., formalized quality systems, the theoretical quality concepts, and service quality tools were reviewed. Similarly, this work categorized smart mobility into innovative solutions and the development of current services. The authors conclude that the consideration of these quality elements has significantly improved the efficiency of service planning and implementation. However, such elements are complex which require measurement tools, and proper planning.

A summary of the literature, studied for this work, is provided in Table 1. From the literature, it is concluded that current solutions focus on the provision of services and application in ITS. However, very little attention is given to the use of software architecture and techniques, even though consideration of software architecture for providing such solutions is very important and useful.

## 3 Research methodology

The main objective of this research is to propose an architecture for providing safety and non-safety ITS services in VANET. To achieve this goal, we followed a systematic approach. The overall research methodology adopted for this work is depicted in Fig. 2. Our proposed methodology starts with the literature review. During this step, we performed a systematic literature review.

We studied different approaches for providing services in VANET. In addition, we also highlight the main limitations of the existing approaches. The second phase of our research methodology is related to the development of software architecture. During this phase, we studied SOA concepts and design constraints. Design constraints are the conditions that are imposed during the development of an architecture. After the development of software architecture, an important activity to perform is the description of the architecture. We described our architecture using UML sequence diagrams. In addition, we considered different case studies to show the applicability of our solution. The use cases selected for this work are pothole detection, accident alert, traffic control system, emergency vehicle warning system, and service advertisement. The final step of our research methodology is the evaluation of the proposed architecture. We evaluated the performance in terms of various design metrics. Furthermore, we also compare the results with smart city architecture.

## 4 Proposed software-oriented architecture

In Section 4, we present our proposed SOA for providing ITS services in VANET. The architecture consists of three layers, which are explained in Subsections 4.1 to 4.8.

**Table 1** Summary of the literature

| Study | Main characteristics | Technique used | Contribution | Limitations |
|---|---|---|---|---|
| Moazzam et al. (2013) | Proposed a road health monitoring system which makes use of cameras for taking images of the pothole. | Image based technique | A low-cost model to determine the depth of a pothole. | Weather and environmental conditions (e.g., heavy rain, visibility, fog) would have a large impact on the quality. Also need complex algorithms for accurate image analysis. |
| Youquan et al. (2011) | Proposed a LED and CCD based technique to capture and analyze the pavement images. | | A digital image processing-based model for analyzing the depth of the potholes. | |
| Lin and Liu (2010) | Proposed an SVM-based solution to distinguish potholes from cracks, and speed breakers etc. | | A method to train the SVM with a set of images. | The training model fails to detect pavement defects if the images are not properly illuminated. |
| Strutu et al. (2013) | Proposed a system which makes use of accelerometer, and GPS sensors. | Sensors based technique | A system architecture in which vehicles are equipped with a local computer, GPS device, and accelerometer. | Providing 100% communication coverage is infeasible. Also the solution is quite expensive and laborious. |
| Singh et al. (2018) | Proposed an ultrasonic sensor-based solution to recognize bumps, potholes, and speed breakers. | | A mobile application was developed which provides alerts to the drivers. | No software architecture is mentioned which would make the implementation harder. |
| Arif and Wang (2020) | Proposed VANET-based solution for providing various services. | Provision of services | A cloud-based architecture for providing services in VANET. | No proper attention was given to the software design (e.g,. service composition). |
| Sándor and Csiszár (2015) | Proposed a real-time solution by considering the actual demands and capacity of parking slots, traffic condition, and control strategies. | | An efficient traffic management model for addressing the limitations of the local methods for parking management. | The authors did not perform any simulation to implement, test, and evaluate the performance of their proposed solution. |
| Nagy and Csiszár (2020) | Proposed an efficient methodology for exploring the quality elements in the context of smart cities. | | A gap-model for providing mobility related services by highlighting the quality management concepts. | These elements are complex which requires measurement tools, and proper planning. |

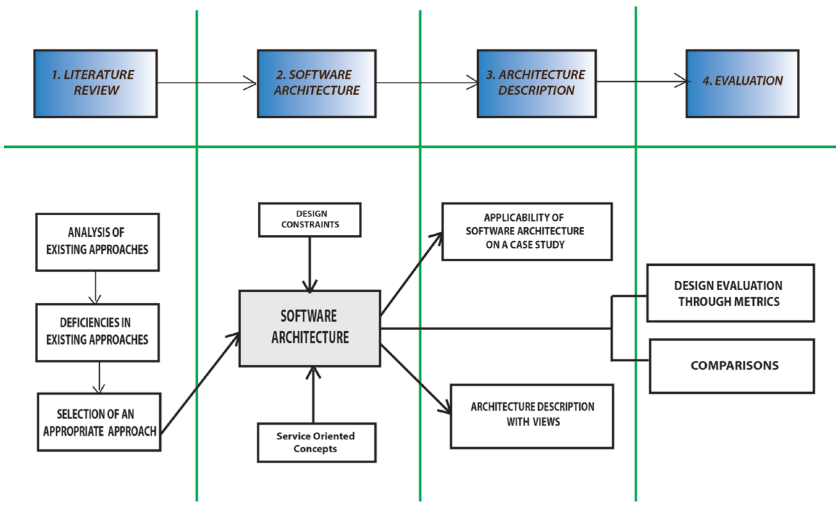


**Fig. 2** Research methodology

### 4.1 Cloud layer

The first layer of the proposed architecture is the cloud layer. The purpose of the cloud layer is to provide a centralized environment for the data collected from vehicles. Any third-party services provider can access this data for providing different services to the user. For example, the road maintenance department has access to the potholes identified by the vehicles.

### 4.2 VANET layer

This core element of the proposed architecture is the VANET layer. Smart vehicles would play an important role in this layer. This layer enables communication among the smart vehicles. It also allows the vehicles to send and receive messages from the RSU. For communication purposes, this layer uses the IEEE 802.11p (Jiang and Delgrossi, 2008).

### 4.3 Application layer

This layer avails the services from the cloud as well as from the RSU. Different applications are possible for deployment at this layer. One such example is an application that enables users to search for the best routes to have a safe journey.

### 4.4 Design constraints

The design constraints are the conditions imposed on a system. These constraints highlight the limitations of the proposed design. The design constraints of the proposed architecture are given in Table 2.

### 4.5 UML stereotypes

UML is one of the well-known general purpose modeling language to visually represent software architecture. UML stereotypes help to model in a domain-specific way. Table 3 provides details about important stereotypes of the proposed solution.

### 4.6 Use cases

The proposed SOA is beneficial for a variety of safety and non-safety ITS applications. To indicate its applicability and usefulness, we have considered several use cases. These use cases are presented in Subsections 4.6.1 to 4.6.5.

#### 4.6.1 Pothole detection

Road health monitoring, especially pothole detection, is a major problem in developing countries. Well-maintained roads contribute a major portion to the country's economy. Real-time detection and reporting of potholes would not only help the drivers to avoid road accidents and vehicle damages but also the authorities. In this use case scenario, the potholes on the surface of the road are detected by smart vehicles. These vehicles are equipped with sensors, GPS devices, and IEEE 802.11p (Jiang and Delgrossi, 2008) based wireless communication antennas. When a pothole is detected the information is communicated to the nearby RSU by using the IEEE 802.11p (Jiang and Delgrossi, 2008) based antenna. The RSU forwards the same information to the cloud. This way the concerned authorities are informed. The sequence diagram for this use case scenario is given in Fig. 3.

#### 4.6.2 Accident alert

When a road accident occurs, a quick emergency response can save people's life. It is important to disseminate the

**Table 2** Design constraints

| S. No | Constraints |
|---|---|
| 1 | Develop an initial architecture |
| 2 | Identify operations of the entire system |
| 3 | Identify services on the basis of identified operations |
| 4 | Separate services into categories |
| 5 | Introduce coordination mechanism for service composition |

**Table 3** UML stereotypes

| Stereotype notation | Concept | Base class | Description |
|---|---|---|---|
| << Cloud >> | Service Provider | Package | Service provider |
| << RSU >> | Service Provider / Consumer | Package | Service provider to provide services to vehicles |
| << Vehicle >> | Service Consumer / Provider | Package | Smart vehicle equipped with various devices |
| << Device >> | Input Provider | Component | Sense the environment |
| << Service >> | Function | Component | Functionality provided by different service providers |
| << Simple Service >> | Simple Service | Component | A single service |
| << Composite Service >> | Composite Service | Component | Group of services |
| << Choreographer >> | Choreographic Service | Component | Services coordinator |

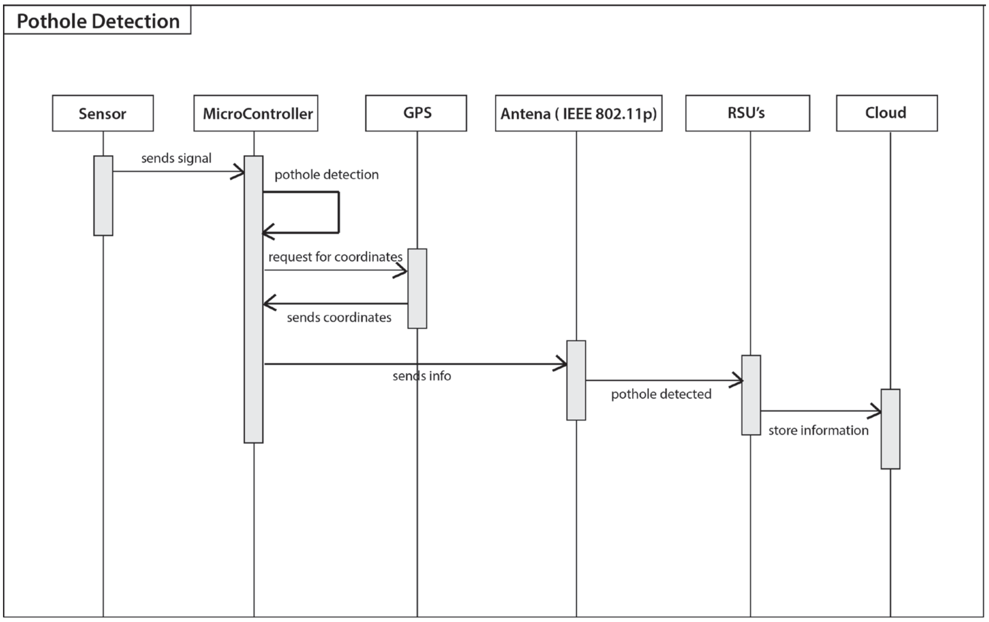


**Fig. 3** Sequence diagram of pothole detection

accident information to the concerned authorities. Both traditional and smart phone-based systems have limitations. For instance, it is hard to explain the exact location of the accident to the telephone operator. In some situations, it is difficult for the victims to access their mobile phones and dial emergency numbers.

In this use case scenario, a vehicle with a built-in camera collects images of a road accident. The collected images are sent to the cloud for analysis purposes. Afterward, the cloud forwards this information to the RSU to alert the drivers about the accident. Such information includes the location of the accident, and alternative routes available. In addition to that, the cloud also sends an alert message to the concerned departments (e.g., police station and hospital). The sequence diagram for this use case scenario is shown in Fig. 4.

### 4.6.3 Traffic control system

The increasing number of vehicles is putting a large pressure on traffic management and controlling authorities. Efficient traffic management systems and policies are required to deal with congested traffic scenarios. In this use case scenario, the RFID tags communicate the traffic-related information with the RSU. Here, the RSU is functioning as an RFID reader. After collecting the information, the

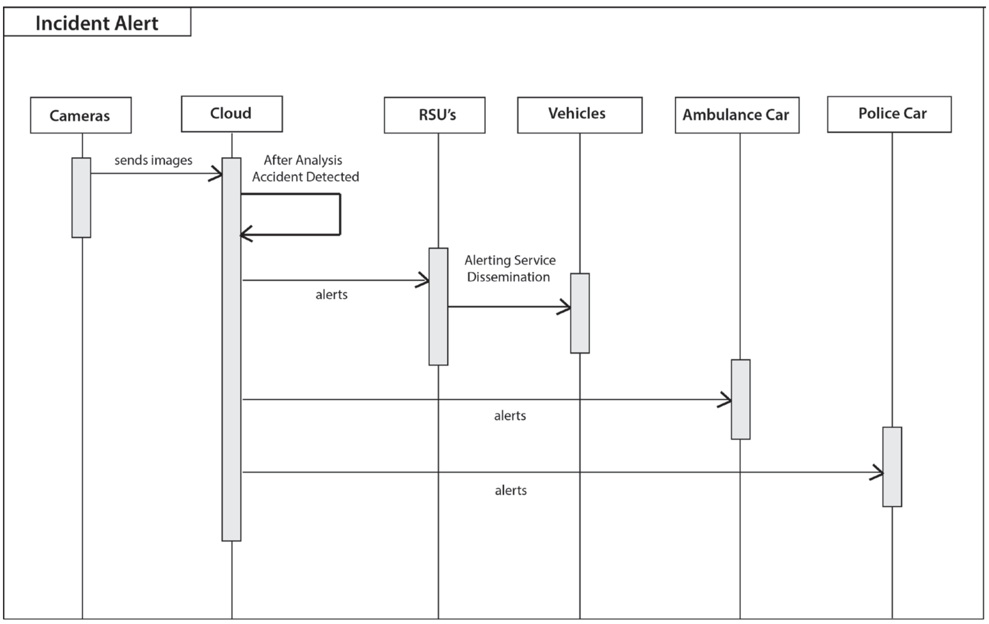


**Fig. 4** Sequence diagram of accident alert

RSU informs the cloud about the current traffic situation in a specific area. This would enable the traffic management and control authorities to take real-time decisions to efficiently manage traffic congestion. The sequence diagram for the traffic control system scenario is shown in Fig. 5.

#### 4.6.4 Emergency vehicle warning system

The traditional methods used by an approaching emergency vehicle are sirens and lights. Such techniques are used to alert drivers to give way to the emergency vehicle. These emergency vehicles try their best to immediately reach the accident location or the hospital. However, several factors, e.g., congestion, and finding the exact location, etc, contribute to the delay involved in providing the emergency services. In this use case scenario, the emergency vehicle will send a message to the nearby RSU. The RSU then controls the traffic lights in such a way that it will allow the emergency vehicle to pass through the intersections without any delay. Furthermore, the RSU alerts the private vehicles to free a lane and give way to the ambulance. The sequence diagram for this use case scenario is shown in Fig. 6.

#### 4.6.5 Service advertisement

Advertisement is a well-known strategy for promoting business and reaching out to a large number of potential customers. Companies use different advertisement campaigns to promote their services. The traditional methods of roadside service discovery (e.g., banners, and billboards) are not effective to attract drivers and passengers. On the other side, the online and mobile advertisement strategies require an Internet connection and are destructive for the drivers. VANET has the potential to promote roadside business services by using the store-carry-and-forward approach.

In this use case scenario, the business owners (e.g., hotel, petrol station, supermarket, restaurant, mechanic workshop, etc.) register their services with a nearby RSU. The RSU then broadcasts the registered services to the smart vehicles. These smart vehicles store the received advertisements and carry them for a predefined time. The drivers who are interested in finding a nearby roadside service sends a query message to the neighboring vehicles. If a vehicle has the required services available, then it will

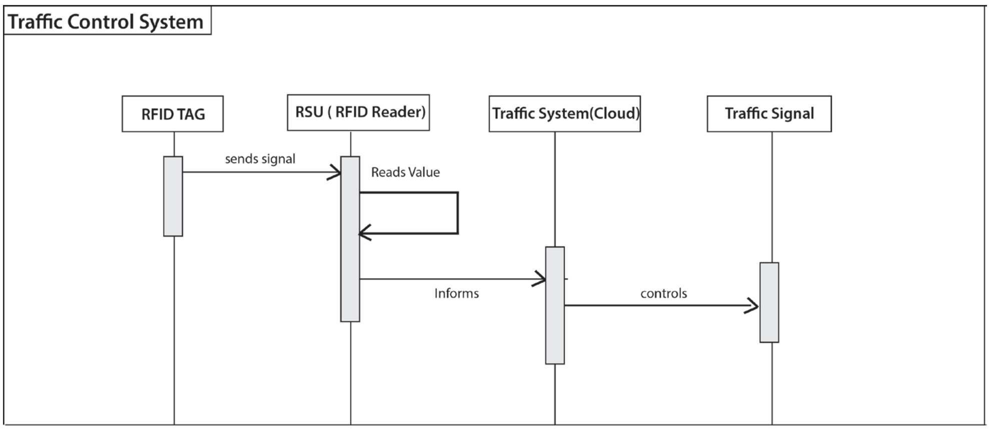


**Fig. 5** Sequence diagram of traffic control system

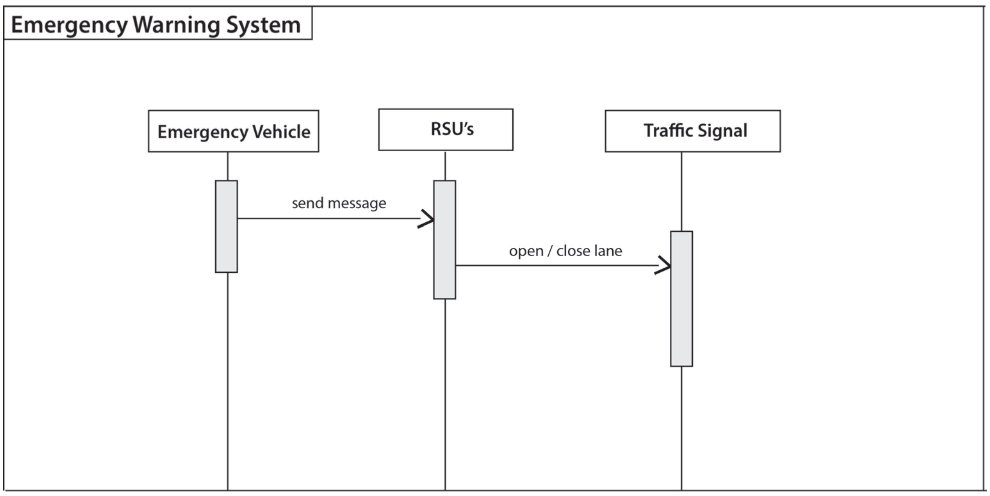


**Fig. 6** Sequence diagram of emergency warning system

respond to the query. The sequence diagram for this use case scenario is depicted in Fig. 7.

### 4.7 Services and operations of the system

The proposed architecture offers different services and related operations. We identify 13 services and their relevant operations from the sequence diagrams of the use cases. These services and their associated operations are listed in Table 4 and explained in Section 4.8.

### 4.8 Categorization of services

After the identification of services, we grouped them into different categories. These categories are explained in the following subsections.

#### 4.8.1 Cloud level services

The services related to the cloud are grouped into this category. These services are listed in Table 5. The first important service in this category is accident detection. The main purpose of this service is to analyze the collected images to detect a road accident. The next service offered by the cloud is called the police service. This service is activated by other services to notify a police station about an unpleasant event. The third service is related to providing emergency services. In case of an emergency, this service is available to other services. The next two services are related to the management of traffic. These services are responsible for the efficient management and control of the traffic, e.g., traffic lights, intersection management, congestion control, and violation of traffic rules, etc. The last service in this category is about providing an interface for the cloud to communicate with the RSU. This interface must be user-friendly and easy to use.

#### 4.8.2 RSU level services

The services offered by an RSU are known as RSU level services. These services are listed in Table 6. The first service in this category is an RFID reader. The RFID reader is responsible for communicating with the RFID tags, mounted inside the vehicles. The RFID tags send vehicle status information at a regular interval. The next service is the traffic signal interface. The main responsibility of this service is to enable communication between the RSU and the traffic control service. This communication is important for efficient traffic management and control. The second last service in this category is called the RSU service advertisement interface. This service provides an opportunity for business owners to disseminate their advertisements for promoting their business. Finally, the last service connects the RSU with the cloud through a user-friendly interface.

#### 4.8.3 Device, business and vehicle level services

The services related to the device, vehicle, and business are grouped into one subcategory. These services are also listed in Table 7. Here, the first service is an example of a device level service. The main responsibility of this service is to send the collected images to the RSU. The RSU then forwards them to the cloud for processing and analysis. The second service in this list is related to the vehicle. This service is responsible for regular broadcasting of the vehicle status messages. These messages are generated by the smart vehicles in response to different events. Finally, the last service is a business level service. It allows the business owner to define different types of advertisement messages for promoting their businesses.

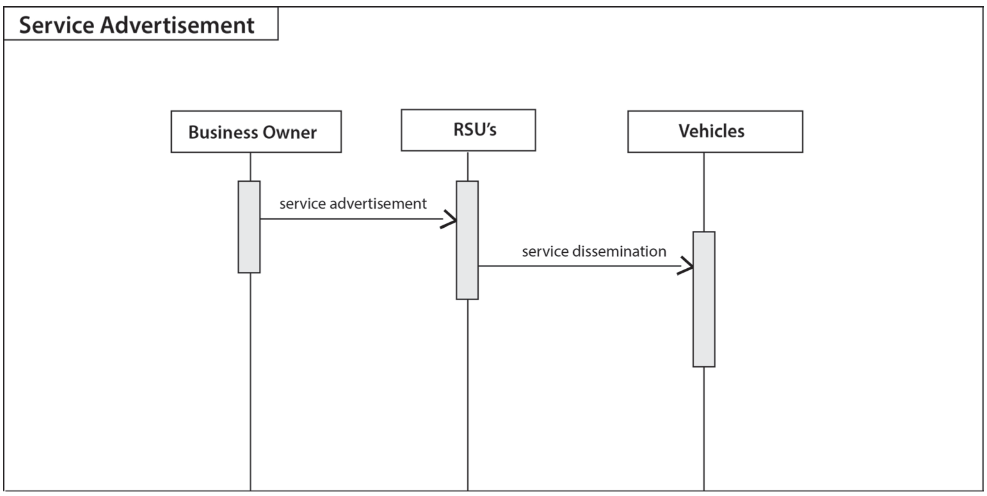


**Fig. 7** Sequence diagram of service advertisement

**Table 4** Services and operations

| S. No | Service | Operations |
|---|---|---|
| 1 | Send / receive message | Send_Message<br>Receive_Message<br>Service_choreography |
| 2 | Send real-time images | Send_Images<br>Service_choreography |
| 3 | Accident detection | Images_Analysis<br>Alert_Police<br>Alert_Ambulance<br>Alert_RSU service advertisement<br>Service_choreography |
| 4 | Police service | Receive_Notification |
| 5 | Ambulance service | Receive_Notification |
| 6 | Traffic control service | Receive_request<br>Operates_traffic_signal<br>Service_choreography |
| 7 | Traffic signal interface | Receive_notification |
| 8 | Traffic violation service | Images_Analysis<br>Alert Police |
| 9 | RFID reader | Reads_value<br>Sends_info<br>Service_choreography |
| 10 | Business service | Advertise_service<br>Service_choreography |
| 11 | RSU interface | Receive_Notification<br>Sends_Info<br>Service_choreography |
| 12 | Cloud interface | Receive_Notification |
| 13 | RSU service advertisement interface | Receive_Notification<br>Service_dissemination<br>Service_choreography |

**Table 5** Cloud level services

| S. No | Service name |
|---|---|
| 1 | Accident detection |
| 2 | Police service |
| 3 | Ambulance service |
| 4 | Traffic control service |
| 5 | Traffic violation service |
| 6 | Cloud interface |

#### 4.8.4 Task oriented services

We also identified some task oriented services. These services are listed in Table 8. The first service in this category is pothole detection. Different sensors installed inside vehicles can detect potholes on the roads. Once a pothole is detected, the vehicle will send this information to the RSU. For this communication, the IEEE 802.11p (Jiang and Delgrossi, 2008) protocol suite is the best option. The next service in this category is the accident alert service. When a road accident is detected by a vehicle, the RSU is informed by the cloud to disseminate the accident-related information (e.g., location, alternative routes, etc.) to the vehicles traveling toward that location. The third service is called emergency vehicle warning. In this case, the emergency vehicle (e.g., ambulance) periodically sends a warning message to the RSU. The RSU controls the traffic

**Table 6** RSU level services

| S. No | Service |
|---|---|
| 1 | RFID reader |
| 2 | RSU signal interface |
| 3 | RSU service advertisement interface |
| 4 | RSU interface |

**Table 7** Device, vehicle and business level services

| S. No | Service |
|---|---|
| 1 | Send images |
| 2 | Send / receive message |
| 3 | Business service |

**Table 8** Task oriented services

| S. No | Services |
|---|---|
| 1 | Pothole detection |
| 2 | Accident alert service |
| 3 | Emergency vehicle warning |
| 4 | Congestion control service |

signals to give way to the ambulance and avoid any delay in providing the emergency services. The last service is related to congestion control. In this scenario, the vehicle sends information (e.g., speed, location, direction, etc.) to the RFID reader. The RFID reader forwards the collected information to the cloud. After analyzing the collected data, the cloud suggests alternative routes to the drivers.

## 5 Results and discussion

In Section 5, we discuss the results of our proposed architecture. To evaluate and compare the performance of our architecture, we used the SOA metrics, as proposed in (Elhag and Mohamad, 2015). The design metrics that we considered are coupling, cohesion, complexity, and reusability. We evaluate the performance of our proposed solution and compare the results with a state-of-the-art software architecture called smart city platform (Kuryazov et al., 2019). The smart city platform is a service-oriented layered architecture, which is developed for the development of smart city applications. For comparison purposes, the communication mechanism between the services of smart city architecture is adopted and the results are calculated on the basis of services collaboration.

### 5.1 Number of services and operations

First, we compare the total number of services and operations of the proposed architecture with the smart city architecture. The results of this comparison are given in Table 9. The total number of services and operations in our architecture are 13 and 30 respectively. On the other hand, the number of services and operations in smart city architecture are 14 and 26 respectively.

### 5.2 Coupling factor

The design quality of software architecture is measured in terms of coupling. The term coupling is used to find the degree of inter-dependencies among different components. A low coupling value is desirable which indicates a good software solution. The coupling of a system could be either direct or indirect. Direct coupling refers to the direct interaction between the entities. On the other hand, an indirect coupling means no direct interactions between the entities. We measured both direct and indirect coupling for our proposed architecture. The results of the coupling metric are shown in Fig. 8. From the results, it is concluded that some of the services, e.g., send/received messages, police service, traffic signal interface, and RSU service

**Table 9** Total number of services and operations

| Software architecture | No of services | No of operations |
|---|---|---|
| Proposed | 13 | 30 |
| Smart city | 14 | 26 |

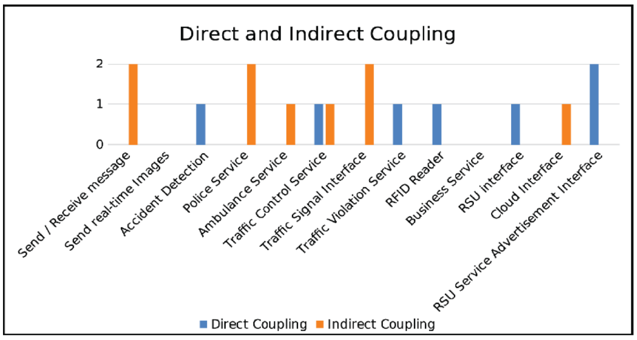


**Fig. 8** Direct and indirect coupling of proposed architecture

advertisement interface have a high degree of interdependence. Furthermore, the results reveal that sending real-time images, and business service have very low inter-dependencies on other components. These services can be offered to the user as a Quality of Service (QoS).

To emphasize the efficiency of our proposed architecture, we compare the results of the coupling factor with the smart city architecture. The results of this comparison are shown in Fig. 9. From the results, it is clear that the coupling factor of our proposed architecture is much smaller than that of the smart city. As a low value is desirable, we conclude that our proposed architecture is more efficient in terms of inter-dependencies.

### 5.3 Cohesion factor

Like coupling, cohesion is also an important design quality metrics. It is the degree of relationship between the elements of a module. Unlike the coupling, a high value of cohesion indicates a good software design. We compare the cohesion factor of the services offered by the proposed and smart city architecture. The results of the comparison are shown in Fig. 10. The results reveal that the proposed architecture has a high cohesion factor for almost all the offered services. The only service for which smart city architecture has a high cohesion value is that orchestrator. However, our proposed architecture does not offer this service.

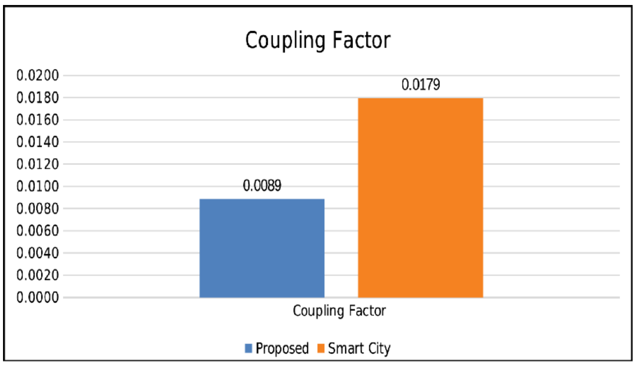


**Fig. 9** Comparison of coupling factor

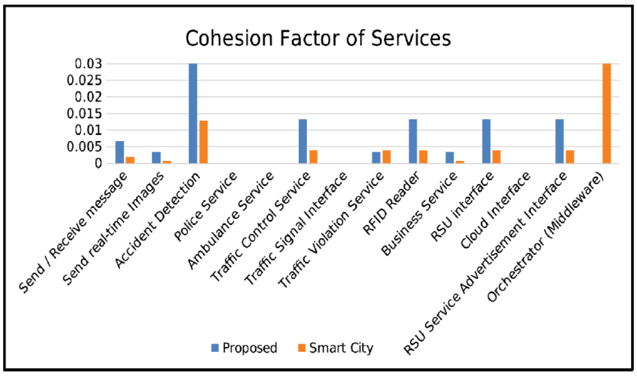


**Fig. 10** Comparison of cohesion factor for individual services

To get more insights, we measure the cohesion factor for the whole system and compare the results with the smart city architecture. The comparative results are shown in Fig. 11. From the results, it is cleared that the proposed architecture is more cohesive as compared to the smart city architecture. The cohesion factor of the proposed architecture is 0.1030, while the value of cohesion for the smart city architecture is 0.737. As a high value is desirable for the cohesion factor, we concluded that our proposed architecture is more efficient in terms of the relationship among the elements of its modules.

### 5.4 Complexity

The quality of software architecture is also measured with the help of complexity metrics. It is another important parameter that calculates the complexity of the service. from coupling and cohesion metrics. First, we measure the complexities of both the architectures in terms of the complex services they offer. These results are given in Fig. 12.

The results show that the complex services offered by our proposed architecture are 7, while the complex services offered by the smart city architecture are 8. The results conclude that the complexity factor of both the architectures is almost the same.

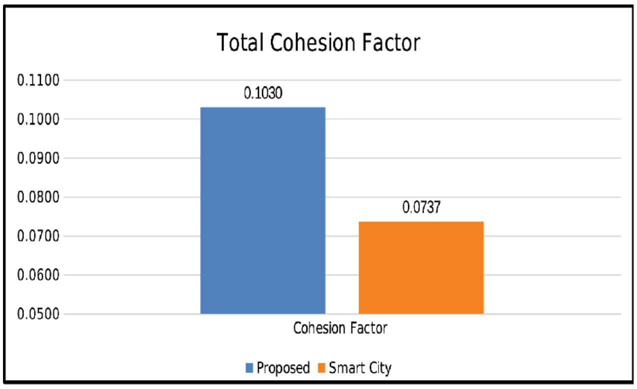


**Fig. 11** Comparison of cohesion factor for the whole system

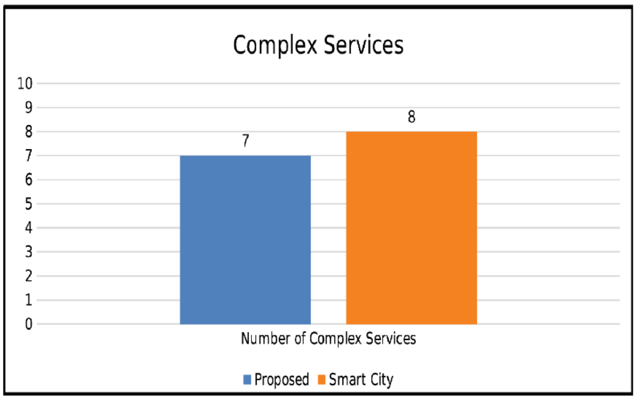


**Fig. 12** Comparison of complex services

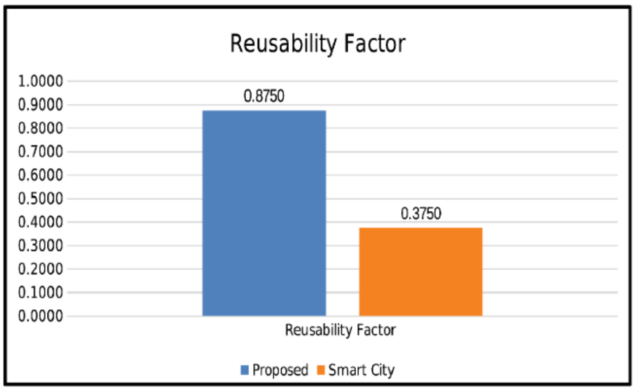


**Fig. 14** Comparison of reusability factor

We also measure the total complexity of our architecture and compare it with smart city architecture. The total complexity of our proposed solution is 26.25, while the overall complexity of the smart city architecture is much higher. These results are plotted in Fig. 13. From the results, we conclude that our proposed architecture is less complex as compared to the smart city. One of the reasons for this improved performance is that smart city architecture depends on the middle-ware. This middle-ware actually increases the coupling and as a result the complexity of the architectures.

### 5.5 Reusability

Software reusability allows the existing software modules or components to be reused without any modification.

It is another important factor for measuring the design quality of architecture. We measured the reusability factor of our proposed architecture and compared it with the smart city. The results of the reusability factor are shown in Fig. 14. From the results, it is clear that the reusability factor of our proposed system is more than the smart city architecture. Here, the reusability factor of our architecture is 0.8750, while it is 0.3750 for the smart city architecture. The results conclude that our architecture is more reusable.

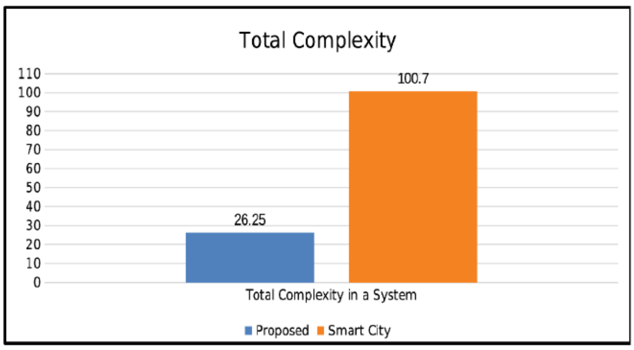


**Fig. 13** Comparison of total complexity

The components and operations of the proposed architecture are easily extendable to provide other types of ITS services.

## 6 Conclusion and future work

Smart transportation will play an important role in future smart cities. VANET paves the way to fulfill the needs of smart and intelligent transportation, It is the cutting edge technology for ITS. A large number of applications have been proposed for VANET. The provision of safety-related services on roads is one of the key aspects of VANET. However, very few solutions consider the use of software engineering principles for their design. We proposed an SOA for providing different services in VANET. As the life and quality of software depend upon the architecture, we believe that our proposed solution is helpful in implementing the futuristic vision of ITS, in the smart city context. We consider several use cases to highlight the scalability of the proposed architecture. We evaluate and compare the performance of our proposed architecture with a state-of-the-art solution by considering different design metrics. From the results, we conclude that our architecture is more suitable for the implementation of the futuristic vision of ITS and VANET.

As future work, we plan to explore the possibilities of integrating the proposed architecture for enhancing the efficiency of safety-related applications. We also intend to implement, test, and evaluate the performance of our proposed architecture by utilizing simulators. For this purpose, we will make use of OMNET++ (Varga and Hornig, 2008), SUMO (Behrisch et al., 2011), and Veins (Sommer et al., 2011) simulators. These simulators are open source and extensively used by research communities, working on VANET and smart cities.

**Acknowledgment**

This research is part of the INCT of the Future Internet for Smart Cities funded by CNPq proc. 465446/2014-0, Coordenacao de Aperfeicoamento de Pessoal de Nivel Superior – Brasil (CAPES) – Finance Code 001, FAPESP proc. 14/50937-1, and FAPESP proc. 15/24485-9. It is also part of FAPESP proc. 21/10647-8.

## References

Arif, M., Wang, G. (2020) "Cloud-based service oriented architecture for social vehicular ad hoc network communications", International Journal of Communication Networks and Distributed Systems, 24(2), pp. 143–166. https://doi.org/10.1504/IJCNDS.2020.104746

Behrisch, M., Bieker, L., Erdmann, J., Krajzewicz, D. (2011) "SUMO – Simulation of Urban MObility: An Overview", In: Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation, Barcelona, Spain, pp. 1–6. [online] Available at: https://elib.dlr.de/71460/1/SUMO_survey_SIMUL2011.pdf [Accessed: 20 April 2021]

Elhag, A. A. M., Mohamad, R. (2015) "Service-oriented design measurement and theoretical validation", Jurnal Teknologi, 77(9), pp. 1–14. https://doi.org/10.11113/jt.v77.6181

Jiang, D., Delgrossi, L. (2008) "IEEE 802.11p: Towards an International Standard for Wireless Access in Vehicular Environments", In: VTC Spring 2008 - IEEE Vehicular Technology Conference, Marina Bay, Singapore, pp. 2036–2040. ISBN 978-1-4244-1644-8 https://doi.org/10.1109/VETECS.2008.458

Khan, S., Babar, M., Khan, F., Arif, F., Tahir, M. (2016) "Collaboration Methodology for Integrating Non-Functional Requirements in Architecture", Journal of Applied Environmental and Biological Sciences, 6(4S), pp. 63–67. [online] Available at: https://www.textroad.com/pdf/JAEBS/J.%20Appl.%20Environ.%20Biol.%20Sci.,%206(4S)63-67,%202016.pdf [Accessed: 25 September 2022]

Kuryazov, D., Khujamuratov, B., Sherkhanov, K. (2019) "Sustainable Service-Oriented Architecture for Smart City Development", In: 2019 International Conference on Information Science and Communications Technologies (ICISCT), Tashkent, Uzbekistan, pp. 1–5. ISBN 978-1-7281-2565-7 https://doi.org/10.1109/ICISCT47635.2019.9011877

Lin, J., Liu, Y. (2010) "Potholes Detection Based on SVM in the Pavement Distress Image", In: 2010 Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science, Hong Kong, China, pp. 544–547. ISBN 978-1-4244-7539-1 https://doi.org/10.1109/DCABES.2010.115

Moazzam, I., Kamal, K., Mathavan, S., Usman, S., Rahman, M. (2013) "Metrology and visualization of potholes using the microsoft kinect sensor", In: 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013), The Hague, Netherlands, pp. 1284–1291. ISBN 978-1-4799-2914-6 https://doi.org/10.1109/ITSC.2013.6728408

Nagy, S., Csiszár, C. (2020) "The quality of smart mobility: a systematic review", Scientific Journal of Silesian University of Technology. Series Transport, 109, pp. 117–127. https://doi.org/10.20858/sjsutst.2020.109.11

Sándor, Z. P., Csiszár, C. (2015) "Role of Integrated Parking Information System in Traffic Management", Periodica Polytechnica Civil Engineering, 59(3), pp. 327–336. https://doi.org/10.3311/PPci.7361

Singh, G., Kumar, R., Kashtriya, P. (2018) "Detection of Potholes and Speed Breaker on Road", In: 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), Jalandhar, India, pp. 490–495. ISBN 978-1-5386-6374-5 https://doi.org/10.1109/ICSCCC.2018.8703298

Sommer, C., German, R., Dressler, F. (2011) "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis", IEEE Transactions on Mobile Computing, 10(1), pp. 3–15. https://doi.org/10.1109/TMC.2010.133

Strutu, M., Stamatescu, G., Popescu, D. (2013) "A mobile sensor network based road surface monitoring system", In: 2013 17th International Conference on System Theory, Control and Computing (ICSTCC), Sinaia, Romania, pp. 630–634. ISBN 978-1-4799-2228-4 https://doi.org/10.1109/ICSTCC.2013.6689030

Ullah, K., Sayyed, A., Aziz, M., Moreira, E. (2019) "A beaconing-based roadside services discovery protocol for vehicular ad hoc networks", Turkish Journal of Electrical Engineering and Computer Sciences, 27(3), pp. 2036–2051. https://doi.org/10.3906/elk-1802-175

Varga, A., Hornig, R. (2008) "An Overview of the OMNeT++ Simulation Environment", In: Simutools '08: Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops, Marseille, France, 60. ISBN 978-963-9799-20-2

Woods, E. (2016) "Software Architecture in a Changing World", IEEE Software, 33(6), pp. 94–97. https://doi.org/10.1109/MS.2016.149

Youquan, H., Jian, W., Hanxing, Q., Zhang, W., Jianfang, X. (2011) "A research of pavement potholes detection based on three-dimensional projection transformation", In: 2011 4th International Congress on Image and Signal Processing, Shanghai, Shanghai, pp. 1805–1808. https://doi.org/10.1109/CISP.2011.6100646