IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received September 21, 2019, accepted October 15, 2019, date of publication November 1, 2019, date of current version November 13, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2950950

Towards a Fog Enabled Efficient Car Parking Architecture

KAMRAN SATTAR AWAISI[1], ASSAD ABBAS[1], MAHDI ZAREEI[2], (Member, IEEE), HASAN ALI KHATTAK[1], (Senior Member, IEEE), MUHAMMAD USMAN SHAHID KHAN[3], MAZHAR ALI[3], IKRAM UD DIN[4], (Senior Member, IEEE), AND SAJID SHAH[3]

[1]Department of Computer Sciences, COMSATS University Islamabad, Islamabad Campus, Islamabad 44500, Pakistan
[2]Tecnologico de Monterrey, Escuela de Ingenieria y Ciencias, Monterrey 64000, Mexico
[3]Department of Computer Sciences, COMSATS University Islamabad, Abbottabad Campus, Abbotabad 22000, Pakistan
[4]Department of Information Technology, The University of Haripur, Haripur 22620, Pakistan

Corresponding authors: Assad Abbas (assadabbas@comsats.edu.pk) and Mahdi Zareei (m.zareei@ieee.org)

This work was supported in part by SEP-CONACyT Research Project under Grant 255387, and in part by the School of Engineering and Sciences and the Telecommunications Research Group, Tecnologico de Monterrey, Monterrey, Mexico.

ABSTRACT The automotive industry is growing day by day and personal vehicles have become a significant transportation resource now. With the rise in private transportation vehicles, getting a free space for parking one's car, especially in populated areas, has not only become difficult but also results in several issues, such as: (i) traffic congestion, (ii) wastage of time, (iii) environmental pollution, and most importantly (iv) unnecessary fuel consumption. On the other hand, car parking spaces in urban areas are not increasing at the same rate as the vehicles on roads. Therefore, smart car parking systems have become an essential need to address the issues mentioned above. Several researchers have attempted to automate the parking space allocation by utilizing state-of-the-art technologies. Significant work has been done in the domains of Wireless Sensor Networks (WSN), Cloud Computing, Fog Computing, and Internet of Things (IoT) to facilitate the advancements in smart parking services. Few researchers have proposed methods for smart car parking using the cloud computing infrastructures. However, latency is a significant concern in cloud-based applications, including intelligent transportation and especially in smart car parking systems. Fog computing, bringing the cloud computing resources in proximate vicinity to the network edge, overcomes not only the latency issue but also provides significant improvements, such as on-demand scaling, resource mobility, and security. The primary motivation to employ fog computing in the proposed approach is to minimize the latency as well as network usage in the overall smart car parking system. For demonstrating the effectiveness of the proposed approach for reducing the lag and network usage, simulations have been performed in iFogSim and the results have been compared with that of the cloud-based deployment of the smart car parking system. Experimental results exhibit that the proposed fog-based implementation of the efficient parking system minimizes latency significantly. It is also observed that the proposed fog-based implementation reduces the overall network usage in contrast to the cloud-based deployment of the smart car parking.

INDEX TERMS Fog computing, smart car parking, fog-based smart car parking, image processing.

I. INTRODUCTION

Rapid advancements in the information and communication technology are urging people to move to urban areas, and consequently, the cities have become overpopulated. Due to the migration of a significant population to cities, the number of vehicles used for commuting daily has increased enormously [1]. Therefore, the parking spaces in large cities have shrunk due to the growing number of vehicles. In this situation, it has become challenging for the drivers to find the car parking slots in peak hours in populated cities. More vehicles pursuing the same parking space creates traffic congestion. Consequently, people spend a lot of time finding the place for parking, which results in wastage of time, added fuel consumption, and environmental pollution. The authors in [1] state that, according to an estimate finding the car

The associate editor coordinating the review of this manuscript and approving it for publication was Christos Verikoukis.

parking space in Loss Angles costs around 730 tons of CO2, 95,000 hours, and 47,000 gallons of gasoline. The population of the world is rapidly urbanizing, and the increasing number of automobiles will add to the traffic congestion problem [2].

Parking problems have enticed more consideration in the last few years due to significant increase in the number of vehicles [3]. Considering the issues and accompanying problems in finding car parking spaces in large cities, several researchers have proposed smart car parking systems using cloud computing and the Internet of Things (IoT). Although the above-mentioned smart car parking systems equipped with the state-of-the-art technologies have offered a new dimension to the research in this domain, their widespread implementation in real-time is still a challenge to be addressed.

One of the possible reasons that hinder widespread acceptability of such systems is the high latency rate and network usage in the current cloud-based implementations. For example, in [4], a cloud-based method is proposed for smart car parking. The authors used the sensors to detect the parking slots and subsequently, the status of the parking slots is updated in the cloud. However, a critical problem in the methodology is timely updating the status in the cloud in case a vehicle enters or leaves a particular parking slot because directly connecting the sensors with the cloud incurs latency which is not tolerable for applications requiring low turnaround time. Therefore, to overcome the shortcomings of cloud based methods in smart car parking systems, we propose the fog based smart car parking architecture. The contributions of this paper are three-fold:

1) Fog-based architecture for the efficient car parking system has been proposed using 3-tiers where fog computing concept is applied at the middle tier. Fog nodes process the parking slot images, and the parking slots information is displayed on the LED. If a particular parking area does not have any vacant parking slot, then the information about nearby parking slots is displayed on the respective parking area's LED.
2) The proposed fog-based architecture considers the latency and network usage factors. Meanwhile, by minimizing the latency and network usage, the driver can choose the parking slot and park the vehicle without wasting time, gasoline, and producing extra carbon dioxide. Therefore, it saves the time for finding the vacant space for parking and waiting time for the parking because the collaboration of the fog nodes gives the information of the nearby vacant parking slots if the parking space is not available in a particular parking area.
3) Simulations are conducted extensively for evaluating the effectiveness and efficiency of the proposed fog enabled car parking architecture. Experimental evaluations demonstrate a significant reduction in latency and network usage as compared to the cloud-based implementation of the smart parking system.

The paper is organized as follows. Section II provides the background information about fog computing whereas Section III discusses the recent research works related to smart car parking systems. Section IV presents the proposed fog-based architecture whereas experimental setup is discussed in Section V. Section VI present the experimental results and discussion. Section VII concludes the paper and also highlights the direction for future research.

II. BACKGROUND

There are several issues associated to the cloud computing, such as higher response time and greater communication costs [5]. Aazam *et al.* [6] compared the cloud and fog computing paradigms and concluded that the fog lowers the processing delay significantly. Fog computing is regarded as an excellent solution to fulfill the demands for large number of connections and low latency applications by performing some computation and processing tasks from the cloud server to the fog node [7]. Fog computing has exhibited a tremendous potential to minimize the latency and network usage issues of cloud computing [8].

Several studies have proposed fog-based architectures in different domains to make their systems more efficient, secure and cost effective. Vilalta *et al.* [9] presented a fog based architecture that provides unified cloud and fog resources for deploying network functions, such as virtualization, mobile edge computing and IoT services at extreme edge of telecommunication operator's network. Bi *et al.* [10] proposed a software-defined-networking based fog computing architecture to support mobility. Moreover, the authors proposed a route optimization algorithm and designed an effective signaling operation to provide transparent and impeccable mobility support to the mobile end users.

A key benefit of fog computing is that it helps in reducing the frequent data transmissions to the cloud. This results in minimizing the network usage required by the applications interacting with the the cloud that eventually results in enhancing the response rate. Many studies have proved that the latency in fog is low as compared to the cloud. For example, in [11], [12] and [13], the authors demonstrated that the latency in fog is low when compared to the cloud-based implementation of the systems. In addition, fog computing reduces the network traffic and supports scalability, and is highly suitable for IoT frameworks [14]. Network usage is always a vital parameter for real-time applications and is effectively minimized using the fog computing [15]. Passas *et al.* [16] have discussed multi radio access technology environment which provide a solution for Paris metro pricing scheme to provide service for customers who avoid to be congested for maintaining the same wagons for different ticket price.

In order to avoid the network congestion problem Miliotis *et al.* [17] propose a weighted proportionally fair bandwidth allocation algorithm for the data volume which is being offloaded thorough WiFi and pricing-based rate allocation is used for the rest of the data volume which is transmitted through LTE uplink.

Fog nodes are geographically distributed and have high computational power. In the geographically distributed architecture of fog computing variety of heterogeneous devices are connected at the end of the network to provide storage, and computation services [18]. Fog provides an increasingly adaptable infrastructure and a better way of processing the data by consuming the low network bandwidth [19]. Therefore, in this paper, we propose to employ fog computing in the smart parking environment.

The proposed smart car parking system provides information about the availability of parking slots using the image processing technique and fog computing upon the entry of a new vehicle in the parking area. The rationale to employ fog computing in the proposed approach is that the information about the available parking slots needs to be updated periodically, therefore, minimizing the latency and network bandwidth consumption is desirable. The issues mentioned above can effectively be dealt with using the fog computing that emphasizes performing computations near the network edges, and consequently, the frequent transmissions of data to the cloud are avoided [20].

In the proposed framework, we employ an image processing technique to determine the availability of a parking slot in the area. The image of the vehicle is captured and subsequently analyzed for various possible positions of a vehicle, such as (i) being parked, (ii) parked, and (iii) leaving the parking area. A Light Emitting Diode (LED) is attached at the entrance of parking to show the availability of vacant slots to the drivers. After the drivers park the vehicle in the slot, the status of that slot is updated in the fog. To evaluate the effectiveness of the proposed smart parking approach, we performed simulations in iFogSim and created scenarios for both the fog-based and the cloud-based implementations of the said system. Experimental results demonstrate that the fog-based implementation of the smart parking environment not only minimizes latency but also reduces the bandwidth consumption as compared to the cloud-based implementation.

III. RELATED WORK

This section discusses the research works that are related to the smart car parking systems.

Typically, the drivers find the parking space by driving around the parking areas, on roadsides, or streets. However, with the advent of technology and smart devices, such as smartphones and surveillance cameras, several automated methods to help drivers park their cars have also been proposed. There are several studies that reveal the smart car parking solutions [21]–[23].

Ji *et al.* [24] proposed a cloud-based smart car parking service for smart cities. The proposed system consists of three-tiers, namely, (i) the application layer, (ii) communication layer, and (iii) sensor layer. The application layer provides the services of the cloud. In the application layer, multiple car parking services, such as car parking locator service, car parking supervision service, and information service, are connected with the information center. The information center provides access to data and the IoT integrated service portal is connected with the IoT management center. The smart city is administered through the IoT integrated service portal. The second layer is the communication layer that enables the communication between the sensor layer and the application layer.

The sensor layer, which is the third layer, consists of RFID, Laser, Infrared, Radar, Ultrasonic, Acoustic, and CCTV cameras. The sensor layer is connected to the communication layer. The communication layer acts as a bridge between the application layer and the sensor layer. The communication layer is dependent on the availability of 3G/4G or Wi-Fi services. However, it is important to mention that the above system is applicable only in smart cities because it heavily relies on sensors. On the other hand, we propose to deploy high definition cameras to capture the images of parking slots, and a microcontroller subsequently transmits the captured images to the fog through a Wi-Fi connection. Moreover, the applicability of our proposed solution is not restricted to the smart cities only.

In [4], the authors proposed a cloud-based architecture for smart car parking. The authors used an Infrared sensor, Raspberry Pi, cloud server, and mobile application to implement the proposed architecture. The system works in a way such that twelve sensors are connected to the Raspberry Pi. The device is registered though the Web API so that it could be considered as part of the system. The value of the slot is detected through the sensor number. Microsoft Azure Web service is used to send the unprocessed and raw data from Web jobs and Web APIs to the Microsoft Azure SQL database.

The data is processed and subsequently used for the parking system. The admin portal is designed to access the data of the parking system to know about the available slots. The portal can also detect the faulty sensors. The data is sent to the mobile application so that the user can access and check the data of parking slots. The limitation in this architecture is that the infrared sensor range is 0.02 meters to 0.4 meters, and the users must have smartphone to access the data. If the users do not have a smartphone, then the system is useless for the user. Moreover, due to the cloud, some issues can be raised, for example, latency as described in [7] and [8]. To increase the computational power and reduce the latency rate fog has been introduced to smart transportation as described in [25]–[27].

The authors in [28] proposed a computer vision based technique that uses Deep Convolutional Neural Network (DCNN) to detect the vacant slot in any parking area. Although the technique is useful in identifying the parking slots, it does not present any associated framework that could help drivers in finding the parking slots at the entry points of the parking areas.

In [29], the authors used cloud computing for smart car parking. The parking module uses the Ultrasonic sensor to detect the vehicle. A mobile phone application is developed for the user to use this system. The sensor determines the status of a car parking slot, and an algorithm finds the nearest parking place according to the user's current location. When the user logs on to the system, the system determines the current location of the user and sends the request to the server. If the system detects the vacant parking space, then the details are provided to the user. The user reserves the car parking space, and the system marks the parking slot as reserved. The limitation in this system is that as described in [28], if the car is parked in a proper way, only then the sensor will be able to identify the presence of the car. Otherwise the sensor will not detect the vehicle, and the system will mark the parking slot as "Available."

An IoT based smart car parking system is presented in [30]. The proposed architecture consists of sensor nodes, fog and cloud nodes and a mobile application. The sensor node operates on 0.5 volts and detects the obstacles. If the obstacle is detected, the output is transmitted to the fog node. However, the problem with this approach is that the driver must have a smart Android phone to get information about the vacant slots in the parking area. The system uses the sensor nodes, and efficiency issues arise due to the inherent limitations of sensor nodes. Moreover, a problem with the approach is that the sensor is able to get details about the obstacle only if the vehicle is parked properly. Furthermore, the efficiency of the sensors depends upon the position and pose of the vehicle. On the other hand, the image processing techniques work differently and efficiently in such environments because of their easy deployment.

In the technique presented in [31], the cameras take the images and extract the features of the images. In case there is an object placed in the slot that resembles a car, then this specific parking slot is considered as occupied. The image segmentation technique is applied to the acquired image, and subsequently, the image is enhanced, and the parking slot is detected. It is essential to mention that we are employing the same procedure as used in [31] for the detection of vacant slots in our proposed architecture.

iFogSim is an open-source tool that helps to simulate the cloud and fog environments [32]. Several research works have used the iFogSim simulator to evaluate their fog based applications' performance. Shurman and Aljarah [33] used a collaborative approach against the baseline approach in the iFogSim toolkit to get results of network usage and total delays. Zohora *et al.* [34] used iFogSim to design FogDevices, Sensor, PhysicalTopology, and Actuator classes. The study in [35] created the simulation scenario in iFogSim to produce an initial investigation into how the EXEGESIS edge computes impacts the amount of data moving throughout a network. Naranjo *et al.* [36] evaluated and compared the performances of the FOCON platform by making simulation in iFogsim. Mahmud *et al.* [37] have described that out of the very few simulation tools available for fog computing, iFogSim is considered as one of the effective tools that offers better support for simulating the fog environments. The authors in [38] used the iFogSim simulator to create a simulation model to demonstrate the proof of concept to calculate the energy-delay performance. In [39], the authors used the iFogSim simulator to test different strategies like concurrent strategy, and First Come-First Served (FCFS) strategy for introducing delay-priority.

IV. PROPOSED SYSTEM ARCHITECTURE

The proposed architecture consists of three layers as depicted in Figure 1. The first layer contains the cameras above the parking lanes or slots and is responsible for capturing the images of parking slots and determining whether parking slot is occupied or not. The second layer of the system is a fog node which is connected to the cameras through a microcontroller device. The third layer contains a cloud that is connected to fog and is responsible for storing and managing the images data for more extended periods. When the driver reaches in front of the parking area, the LED screen displays the status of parking slots to the driver. The proposed framework does not require the smartphone to verify the information about the vacant slots in the parking area. Therefore, to minimize the traffic congestion in the parking areas and wastage of fuel, the drivers are directed toward the parking slot at the entrance gate of parking and park the vehicle.

A. OVERVIEW

The proposed fog-based architecture consists of various devices, such as cameras, microcontroller chip, fog node, smart LED, and the cloud server. The cameras in the proposed architecture are responsible for capturing the images of the parking slots. Subsequently, based on the captured images, the availability of the parking slots is determined using the image processing technique presented in [31]. The fog node acquires the pictures and updates the display of the parking status every five seconds.

Multiple cameras are placed in the parking area to cover all the parking lanes. In the proposed architecture, the microcontroller acts as the bridge between the fog node and the cameras. The fog node communicates with the cloud, and the data from the fog node is transferred to the cloud after a certain period. Different fog nodes deployed at different parking spaces, cooperate to share the parking space information. If there is no vacant parking slot in a particular parking area, then the fog node can fetch the information of parking slots from nearby parking slots and information is displayed on the LEDs.

Cloud computing has the capacity to process and manage the data for a longer time; however, frequent access to the cloud to transmit and access the data not only increases the latency but also consumes high network bandwidth as a result of which the other applications also suffer. By introducing the middle layer of fog node, the latency is decreased because the frequent accesses to the cloud are circumvented as the frequent transmission of images to the cloud, processing the

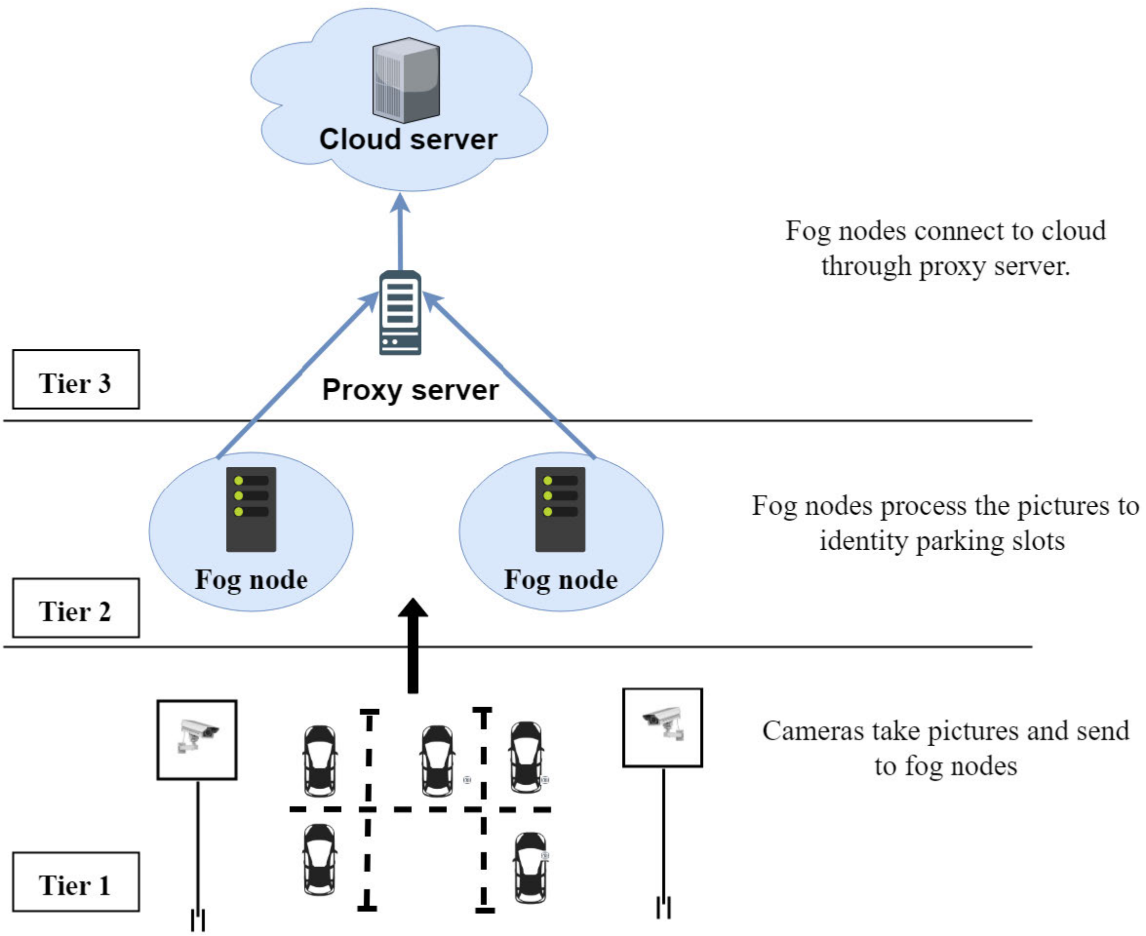


FIGURE 1. **Three tier architecture for Fog enabled Smart Parking system equipped with cameras and sensors.**

images to determine the available parking slots, and retrieval of information on the LED are time-consuming tasks as compared to fog. In the presented architecture, the fog node and the cloud have two-way communication and the tasks of processing and storing the images take place at the fog node. The image data is transmitted to the cloud after a particular time to store for a extended period of time.

The architecture of the proposed fog-based car parking system for one parking area and multiple parking areas are presented in Figures 2 and 3, respectively. As shown in Figure 2, if there is only one parking area, then only one fog node will be deployed in that specific area, and the fog node will send the data to the cloud server for long time usage. On the other hand, Figure 3 depicts the architecture for multiple parking areas and fog nodes are connected to one cloud server. Every parking area will have its own fog node that will be connected to the centralized cloud server. In this case, the latency and network usage will remain the same for every fog node, but the time and network usage to upload the data and to retrieve the data from a centralized cloud server will increase. The components of the proposed architecture are discussed below:

1) CAMERAS AND MICROCONTROLLER LAYER

The cameras and microcontroller are the tier-1 of the smart car parking system. We preferred cameras over sensors to detect the vacant parking spaces for the reason that sensors rely heavily on the position of the parked vehicle while the image processing based approach identify and locate the vacant parking slot on the basis of parking line segments [28]. Our main focus in this study is to compare the latency rate and network usage of fog and cloud servers.

The cameras capture the images and the group of cameras are connected to the microcontroller. The cameras employed in the proposed framework use 12 Volts. The microcontroller is programmed in such a way that it transfers the captured images of parking slots to the fog node for further processing as described in [31].

At first, all the cameras are initialized, and then the image of the parking slots is acquired from the top view through high definition cameras present in the parking space. After the image acquisition phase, the image segmentation is performed where the acquired image is converted into the RGB image. The RGB image is subsequently processed to the grey scale image. On the grey scale image, threshold optimizing

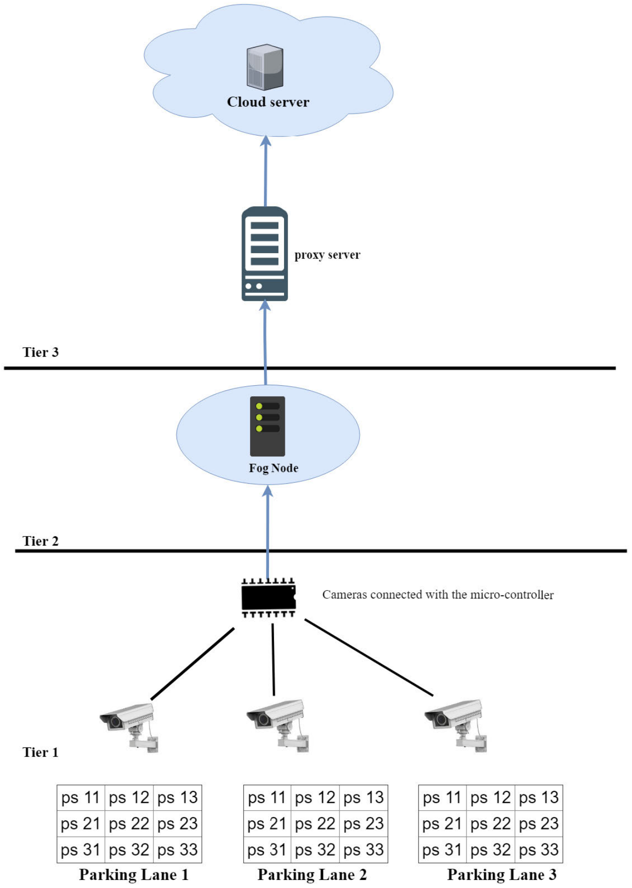


FIGURE 2. **Architecture of the smart car parking system for one parking area.**

techniques are applied, and the final binary image is acquired for segmentation. The binary image is enhanced to remove the unwanted noise, and subsequently, the image is detected to determine whether the parking slot is free or not. The steps for capturing the picture and detecting the slot for parking are presented below:

1) *Initialize the system*
    a) *Set cameras stationary*
2) *Image Acquisition*
    a) *Capture the image of parking area*
3) *Image segmentation*
    a) *Input the RGB image*
    b) *Convert the RGB image into gray scale*
    c) *Apply the threshold techniques*
    d) *Get final image for segmentation*
4) *Apply image enhancement technique*
    a) *Remove the noise from image*
5) *Image detection*
    a) *Extract the features of image*

2) THE FOG NODE

It is the middle layer between the cameras and the cloud and is used to collect the images data from cameras using the microcontroller to further detect the status of the parking slot. The parking slots have specific indexes, for example, ps11 means row 1 and slot 1 in any particular parking lane. The fog updates the parking slot status to the smart LED, which is displayed at the entrance of the parking area. The fog node stores the image data for a specific time and then transfers the data to the cloud. To process the real-time data of parking slots, the fog computing infrastructure is enabled as a supporting middle layer in between edge nodes and the cloud to collect, process, and analyze the data at the edge.

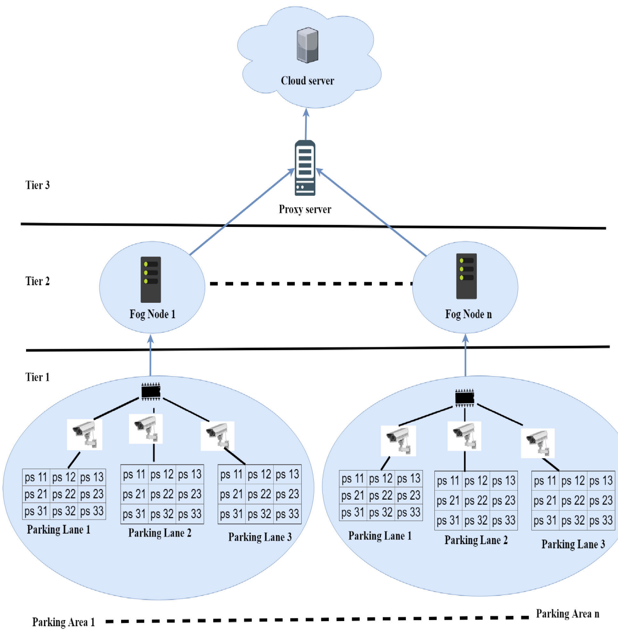


FIGURE 3. **Architecture of smart car parking system for multiple parking areas.**

3) THE CLOUD LAYER

The role of cloud in the proposed framework is that it stores the image data in its storage when it is no more needed by the fog node. The communication between the fog and the cloud is enabled through proxy server. The fog node passes the images data to the cloud after a specific time period, and if the fog needs some image data, then the data is provided by the cloud.

Among many prominent properties of the fog computing is interoperability, which plays important role when we consider the underlying diversity of edge nodes as shown in Figure 4 [40]. Fog nodes extricate some of the resources to coordinate with each other in order to fulfill the computation and storage needs of the neighbouring fog nodes [5]. In our scenario, we assume that fog nodes can collaborate through proxy server and communicate over among the neighbouring nodes the essential information. As interoperablity is the inherent property, therefore we will not take into account the latency for communication of fog nodes. When there is no vacant parking slot in a particular parking space, then the availability of the nearest parking slot will be displayed on the to the requesting node.

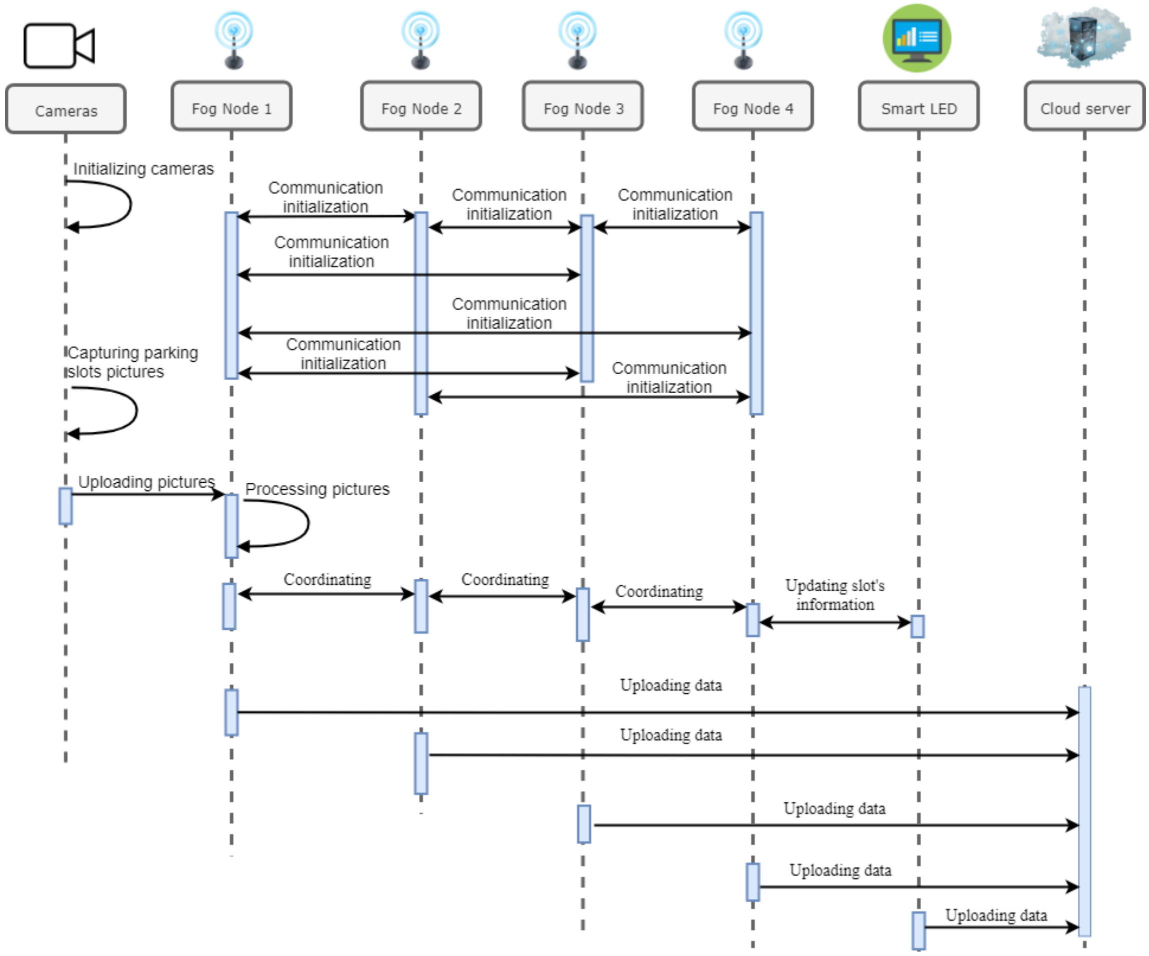


FIGURE 4. Interaction between cameras, fog nodes and cloud data center.

V. EXPERIMENTAL SETUP

We simulated the scenario that involves the high definition smart cameras to capture the images of the parking slots. The images are subsequently transmitted to the fog node. The fog node processes the images to detect the status of the slot and displays the pictures of parking slots on a smart LED connected to the fog node using the Wi-Fi connection. A proxy server establishes the connection between the fog node(s) and the cloud server. To simulate the scenarios, we used iFogSim, which is a toolkit for the IoT devices. By using the iFogSim, we can evaluate the latency, and network usage.

We created the variables of parking areas and a number of cameras in the simulation. We created four parking areas in our experimental scenario. In each parking area, initially four cameras were assigned to capture the images of the parking area. We created a primary cloud device to which our fog nodes are connected using the proxy server. It is essential to mention that for each area, we created one fog node. As the cameras are smart (WiFi enabled) and connected through microcontroller, therefore we created the cameras in the simulation environment and considered them as sensors by the policies of [32]. We increased the number of cameras to evaluate the results for different scenarios. For a particular scenario, we increased the number of cameras to evaluate the effects on the latency, and total network use in a fog node.

Figure 5 represent the topology created in iFogSim to evaluate the fog scenario results. We created four fog nodes in this topology and four cameras were attached to each fog node. This topology was created to evaluate the latency and network usage in iFogSim. We embedded the picture snippet module in cameras to capture the images of the parking slots. We created the slot detector module and embedded it in fog nodes to process the pictures in order to detect vacant parking slots . Moreover, the fog node updates the parking slots status to the connected smart LED.

In comparison to Figure 5, as the number of parking areas increase, the fog nodes are created, and the task of determining the availability of the parking slots are performed on fog nodes. When the number of cameras and LED's increases in a particular fog node, then latency rate and network usage of distinct fog node increases. A benefit of performing the processing task here is that the computation burden on

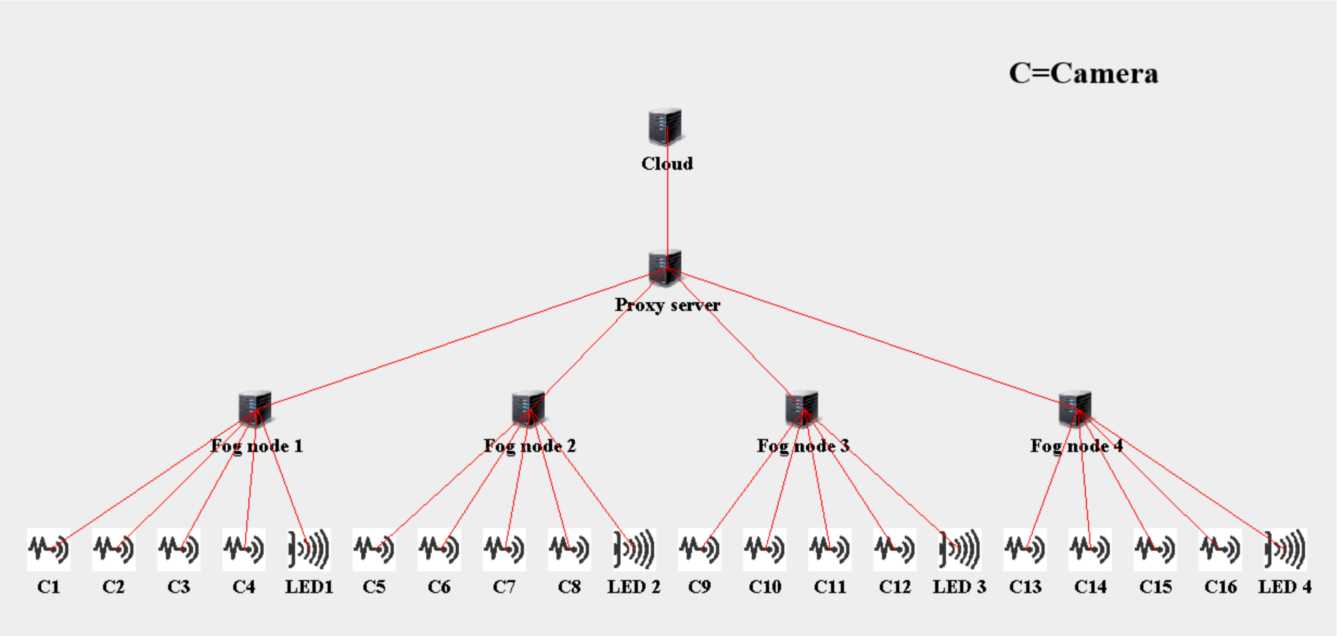


FIGURE 5. iFogSim topology of 16 cameras connected to 4 fog nodes connected to the cloud server via the proxy server.

TABLE 1. Value of parameters of Fog, Proxy and Cloud for fog based scenario.

Parameter	Cloud	Proxy	Fog
CPU length (MIPS)	44800	2800	2800
RAM (MB)	40000	4000	4000
Uplink bandwidth (MB)	100	10000	10000
Downlink bandwidth (MB)	10000	10000	10000
Level	0	1	2
RatePerMIPS	0.01	0.0	0.0
Busy power(Watt)	16*103	107.339	107.339
Idle power (Watt)	16*83.25	83.43	83.43

TABLE 2. Value of parameters of Cloud and Router for cloud based scenario.

Parameter	Cloud	Router
CPU length (MIPS)	44800	2800
RAM (MB)	40000	4000
Uplink bandwidth (MB)	100	10000
Downlink bandwidth (MB)	10000	10000
Level	0	1
RatePerMIPS	0.01	0.0
Busy power(Watt)	16*103	107.339
Idle power (Watt)	16*83.25	83.43

the cloud decreases significantly because the computations are performed on the fog nodes. However, connecting the cameras directly to the cloud server using router results in increased latency and consequent high network bandwidth consumption.

Table 1 presents the configuration parameters of the cloud server, proxy server, and fog server created during the simulation of fog based scenario. The configuration parameters include processing capability in per million instructions,Random Access Memory (RAM),uplink bandwidth, downlink bandwidth,level,rate or cost of per million instruction processing, busy power and idle power.

Figure 6 illustrates the cloud based setup in iFogsim. For the performance evaluation in cloud based scenario multiple cameras and LED are attached to cloud server through router. Cameras post the images of the parking slots to the cloud server. Cloud server process the pictures of parking slots and the information of the parking slots is displayed on the LED which is connected to the cloud server. The number of cameras were gradually increased to examine the value of latency and network usage. Table2 shows the configuration parameters of cloud and router created for the simulation of cloud based scenario.

VI. EXPERIMENTAL RESULTS AND DISCUSSION

This section presents the results of the proposed fog-based architecture in terms of latency and network usage and compares them with the cloud-based implementation. Experimental results demonstrate that the fog-based implementation results in low latency and low network usage as compared to the cloud-based implementation. Table 3 presents the results of latency and network usage in the fog environment along with the results of cloud-based implementation.

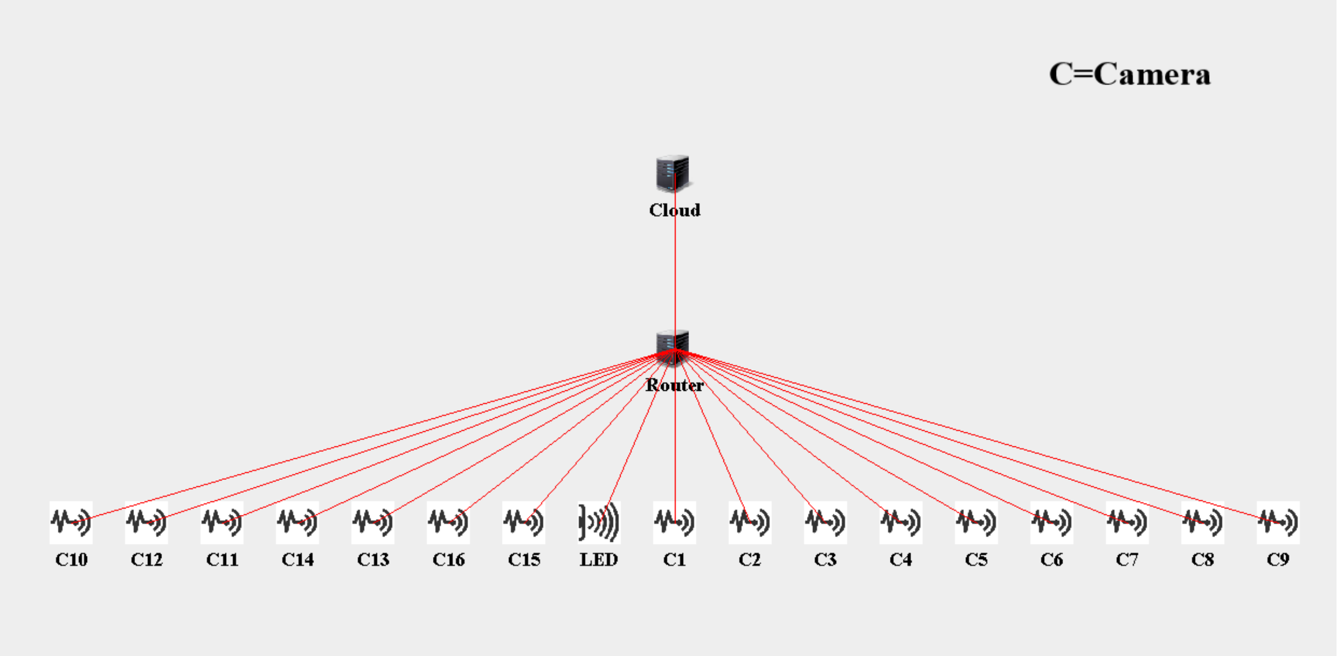


FIGURE 6. iFogSim topology of cloud enabled scenario with 16 cameras connected through a router.

TABLE 3. Simulation results for the proposed car parking architecture for fog and cloud.

Cameras	Fog Latency (ms)	Cloud Latency (ms)	Fog Network usage (kB)	Cloud Network usage (kB)
16	7.87	8.423	3198.4	27632.16
20	8.23	9.5	3998	36099.48
24	8.59	10.73	4797.6	45206.12
28	8.95	12.13	5597.2	54951.48
32	9.30	292.6	6396.8	62398.28
40	10.02	1968.6	7996	63802.54
44	10.3	2489.4	8795.6	64557.22
48	10.73	2886.9	9595.2	64859.64

A. ANALYSIS OF LATENCY

Latency is the necessary factor to be reduced in environments demanding high performance in real-time. A key benefit of the fog computing is that it evades frequent accesses to the cloud and performs computations at the edge of network to offer quick response back to the client device thereby minimizing the latency. The images of the parking slots are transmitted to the fog nodes for processing that are present at the edge of the network. Because one fog node is dedicated to one area, therefore, enough compute power is available to process the images of that area and update the information of available slots on LED's in lesser time. The latency is computed using Eq. 1 which is derived from [32].

$$\text{Latency} = \alpha + \mu + \varphi \tag{1}$$

where α is the Tuple CPU Execution Delay for capturing pictures and μ is the time to upload pictures on fog node for processing and storage. Finally, φ is the time taken to display the information to the LED after processing at the Fog node.

B. ANALYSIS OF NETWORK USAGE

When the traffic increase on cloud server then, only the cloud resources are used. Increase of traffic on the cloud server results in increased network usage. Consequently, the data rates on the network decrease due to the increased traffic. For geographically distributed servers, one fog node is dedicated for one geographical area to deal with the request of that area. Consequently, the network usage in that case decreases and the transmission rate for the rest of the traffic increases. The network usage is calculated using Eq. 2 which is derived from [32].

$$\text{Network usage} = \text{Latency} * \partial \tag{2}$$

where

$$\partial = \text{tupleNWSize}$$

Experimental results demonstrate the effectiveness of our proposed fog-based smart car parking architecture as shown

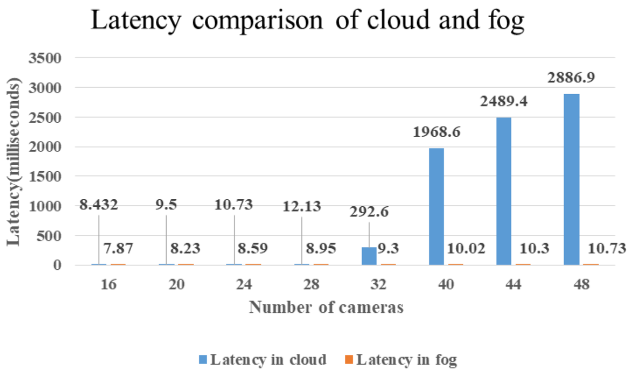


FIGURE 7. Comparison of latency in fog computing and cloud computing based smart parking system.

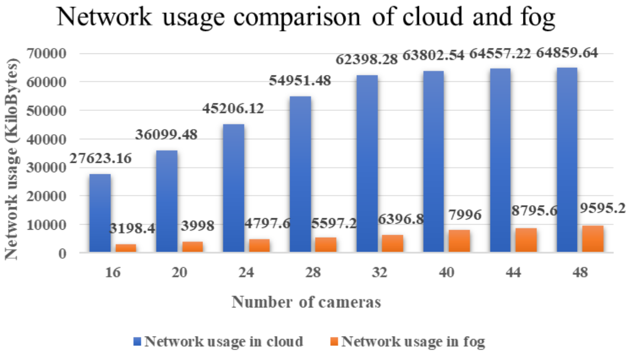


FIGURE 8. Comparison of network usage in fog computing and cloud computing based smart parking system.

in Figure 1. As can be seen in Table 3 that different scenarios for fog and cloud based setup were created where the varying number of cameras attached to fog nodes and cloud server have been realized. In order to evaluate the fog based results we created four fog nodes. For scenario 1, sixteen cameras are attached with 4 fog nodes. Each fog node is connected to the four cameras. Likewise, for all the scenarios the number of cameras are increasing with the fog nodes and cloud server. For a particular scenario, for example scenario 2, if there are twenty cameras, it means that 5 cameras are connected with each fog node.Through out the performance evaluation, the number of the fog nodes remains same in all the scenarios. In case of cloud scenario, we connected the cameras to the cloud through router.

We ran the scenarios of Table 3 in iFogSim to calculate the results. Our main focus for the result was latency and network usage calculation. Figure 7 presents the comparison of latency for both the cloud and the fog environment. From the results, it can be observed that with the increase in number of cameras, the latency in the cloud significantly increases as compared to the fog. The reason is that when the number of cameras are increased in the fog, the fog node dedicated for that particular area processes the images for that area only. On the other hand, in the case of cloud, the cloud server processes the images of all the parking areas and consequently the latency increases when the number of cameras is increased.

Figure 8 demonstrates the results of network usage for fog scenario. It can be seen in Figure 8 that as we increase the number of fog nodes and cameras, the network usage increases. The reason for increased network usage in the cloud environment is that when we connect all the cameras to the cloud server, all the tuples are processed in one cloud server at a time and the network usage increases. On the other hand, in the case of fog, multiple cameras are connected with multiple fog nodes where one fog node is dedicated for one parking area and consequently, one fog node is only responsible to process tuples of those cameras which are deployed in that particular area.

The experimental results for both the fog-based and the cloud-based implementation for the two evaluation parameters namely, the latency and network usage demonstrate the effectiveness of the proposed fog-based architecture for smart parking system.

By using the fog-based architecture for smart car parking, the information about vacant parking slots can be obtained timely in a certain parking area and also it will reduce the time, fuel, and carbon dioxide emission while searching the parking slots. Moreover, the results also help us recognize the potential of fog computing in the IoT settings where shorter turnaround times are highly desirable. In conclusions, low latency and low network usage make the fog based architecture more practical for real-time applications and scenarios.

VII. CONCLUSION AND FUTURE WORK

In recent times, Fog computing has assumed a pivotal role, especially in time-sensitive application domains. Due to the increase in the data generating devices, the need for quicker response has also been intensified. To that end, we proposed a fog-based smart parking architecture that uses computer vision technique to identify an empty parking slot thus enabling drivers to find the parking slot in minimum time that eventually minimizes the time and fuel consumption. The experimental results reveal that the proposed fog-based architecture not only minimizes the latency but also reduces the network usage as compared to the cloud.

A limitation of the proposed research is the use of cameras for parking space detection. This may give rise to the privacy issues for the car owners as the images are stored in the cloud. Considering the fact that most of the storage and processing is handled on the nearby fog nodes. Though, there is a need to preserve the privacy of data in cloud storage by applying suitable encryption techniques and can be an important direction for future work. Moreover, it is also important to mention that the large-scale application of the proposed framework with the increased number of parking areas will require balancing the load on fog nodes to maintain the efficiency. Therefore, in future we intend to investigate the load balancing issues in fog nodes and devise an effective solution to resolve the issue.

REFERENCES

[1] A. Aliedani, S. W. Loke, A. Desai, and P. Desai, "Investigating vehicle-to-vehicle communication for cooperative car parking: The copark approach," in *Proc. IEEE Int. Smart Cities Conf. (ISC2)*, Sep. 2016, pp. 1–8.

[2] A. Khanna and R. Anand, "IoT based smart parking system," in *Proc. Int. Conf. Internet Things Appl. (IOTA)*, Jan. 2016, pp. 266–270.

[3] C. Tang, X. Wei, C. Zhu, W. Chen, and J. J. P. C. Rodrigues, "Towards smart parking based on fog computing," *IEEE Access*, vol. 6, pp. 70172–70185, 2018.

[4] S. Ravishankar and N. Theetharappan, "Cloud connected smart car park," in *Proc. Int. Conf. I-SMAC (IoT Social, Mobile, Anal. Cloud) (I-SMAC)*, Feb. 2017, pp. 71–74.

[5] M. Mukherjee, L. Shu, and D. Wang, "Survey of fog computing: Fundamental, network applications, and research challenges," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 1826–1857, 3rd Quart., 2018.

[6] M. Aazam, S. Zeadally, and K. A. Harras, "Fog computing architecture, evaluation, and future research directions," *IEEE Commun. Mag.*, vol. 56, no. 5, pp. 46–52, May 2018.

[7] Y.-C. P. Chang, S. Chen, T.-J. Wang, and Y. Lee, "Fog computing node system software architecture and potential applications for NB-IoT industry," in *Proc. Int. Comput. Symp. (ICS)*, Dec. 2016, pp. 727–730.

[8] A. Alrawais, A. Alhothaily, C. Hu, and X. Cheng, "Fog computing for the Internet of Things: Security and privacy issues," *IEEE Internet Comput.*, vol. 21, no. 2, pp. 34–42, Mar./Apr. 2017.

[9] R. Vilalta, V. Lopez, A. Giorgetti, S. Peng, V. Orsini, L. Velasco, R. Serral-Gracia, D. Morris, S. De Fina, F. Cugini, P. Castoldi, A. Mayoral, R. Casellas, R. Martinez, C. Verikoukis, and R. Munoz, "TelcoFog: A unified flexible fog and cloud computing architecture for 5G networks," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 36–43, Aug. 2017.

[10] Y. Bi, G. Han, C. Lin, Q. Deng, L. Guo, and F. Li, "Mobility support for fog computing: An SDN approach," *IEEE Commun. Mag.*, vol. 56, no. 5, pp. 53–59, May 2018.

[11] B. K. Dar, M. A. Shah, H. Shahid, and A. Naseem, "Fog computing based automated accident detection and emergency response system using Android smartphone," in *Proc. 14th Int. Conf. Emerg. Technol. (ICET)*, Nov. 2018, pp. 1–6.

[12] K. P. Saharan and A. Kumar, "Fog in comparison to cloud: A survey," *Int. J. Comput. Appl.*, vol. 122, no. 3, pp. 1–4, Jan. 2015.

[13] H. F. Atlam, R. J. Walters, and G. B. Wills, "Fog computing and the Internet of Things: A review," *Big Data Cogn. Comput.*, vol. 2, no. 2, p. 10, 2018.

[14] A. V. Dastjerdi, H. Gupta, R. N. Calheiros, S. K. Ghosh, and R. Buyya, "Fog computing: Principles, architectures, and applications," in *Internet Things*. Amsterdam, The Netherlands: Elsevier, 2016, pp. 61–75.

[15] A. V. Dastjerdi and R. Buyya, "Fog computing: Helping the Internet of Things realize its potential," *Computer*, vol. 49, no. 8, pp. 112–116, Aug. 2016.

[16] V. Passas, V. Miliotis, N. Makris, T. Korakis, and L. Tassiulas, "Paris metro pricing for 5G HetNets," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–6.

[17] V. Miliotis, L. Alonso, and C. Verikoukis, "Weighted proportional fairness and pricing based resource allocation for uplink offloading using IP flow mobility," *Ad Hoc Netw.*, vol. 49, pp. 17–28, Oct. 2016.

[18] A. A. Mutlag, M. K. A. Ghani, N. Arunkumar, M. A. Mohammed, and O. Mohd, "Enabling technologies for fog computing in healthcare IoT systems," *Future Gener. Comput. Syst.*, vol. 90, pp. 62–78, Jan. 2019.

[19] V. Kumar, A. A. Laghari, S. Karim, M. Shakir, and A. A. Brohi, "Comparison of fog computing & cloud computing," *Int. J. Math. Sci. Comput.*, vol. 5, no. 1, pp. 31–41, 2019.

[20] H. A. Khattak, S. U. Islam, I. U. Din, and M. Guizani, "Integrating fog computing with VANETs: A consumer perspective," *IEEE Commun. Standards Mag.*, vol. 3, no. 1, pp. 19–25, Mar. 2019.

[21] T. Lin, H. Rivano, and F. Le Mouël, "A survey of smart parking solutions," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 12, pp. 3229–3253, Dec. 2017.

[22] K. Hassoune, W. Dachry, F. Moutaouakkil, and H. Medromi, "Smart parking systems: A survey," in *Proc. 11th Int. Conf. Intell. Syst., Theories Appl. (SITA)*, Oct. 2016, pp. 1–6.

[23] T. Delot, S. Ilarri, S. Lecomte, and N. Cenerario, "Sharing with caution: Managing parking spaces in vehicular networks," *Mobile Inf. Syst.*, vol. 9, no. 1, pp. 69–98, 2013.

[24] Z. Ji, I. Ganchev, and M. O'Droma, and X. Zhang, "A cloud-based intelligent car parking services for smart cities," in *Proc. 31th URSI Gen. Assem. Sci. Symp. (URSI GASS)*, Aug. 2014, pp. 1–4.

[25] W. Balzano and F. Vitale, "DiG-Park: A smart parking availability searching method using V2V/V2I and DGP-class problem," in *Proc. 31st Int. Conf. Adv. Inf. Netw. Appl. Workshops (WAINA)*, Mar. 2017, pp. 698–703.

[26] T. Wang, J. Zeng, Y. Lai, Y. Cai, H. Tian, Y. Chen, and B. Wang, "Data collection from WSNs to the cloud based on mobile Fog elements," *Future Gener. Comput. Syst.*, to be published.

[27] T. Wang, J. Zhou, M. Huang, M. D. Z. A. Bhuiyan, A. Liu, W. Xu, and M. Xie, "Fog-based storage technology to fight with cyber threat," *Future Gener. Comput. Syst.*, vol. 83, pp. 208–218, Jun. 2018.

[28] L. Zhang, J. Huang, X. Li, and L. Xiong, "Vision-based parking-slot detection: A DCNN-based approach and a large-scale benchmark dataset," *IEEE Trans. Image Process.*, vol. 27, no. 11, pp. 5350–5364, Nov. 2018.

[29] A. Zajam and S. Dholay, "Detecting efficient parking space using smart parking," in *Proc. 9th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Jul. 2018, pp. 1–7.

[30] W. Alsafery, B. Alturki, S. Reiff-Marganiec, and K. Jambi, "Smart car parking system solution for the Internet of Things in smart cities," in *Proc. 1st Int. Conf. Comput. Appl. Inf. Secur. (ICCAIS)*, Apr. 2018, pp. 1–5.

[31] K. Kaarthik, A. Sridevi, and C. Vivek, "Image processing based intelligent parking system," in *Proc. IEEE Int. Conf. Electr., Instrum. Commun. Eng. (ICEICE)*, Apr. 2017, pp. 1–4.

[32] H. Gupta, A. V. Dastjerdi, S. K. Ghosh, and R. Buyya, "iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, edge and fog computing environments," *Softw., Pract. Exper.*, vol. 47, no. 9, pp. 1275–1296, 2017.

[33] M. M. Shurman and M. K. Aljarah, "Collaborative execution of distributed mobile and IoT applications running at the edge," in *Proc. Int. Conf. Electr. Comput. Technol. Appl. (ICECTA)*, Nov. 2017, pp. 1–5.

[34] F. T. Zohora, M. R. R. Khan, M. F. R. Bhuiyan, and A. K. Das, "Enhancing the capabilities of IoT based fog and cloud infrastructures for time sensitive events," in *Proc. Int. Conf. Elect. Eng. Comput. Sci. (ICECOS)*, Aug. 2017, pp. 224–230.

[35] E. K. Markakis, K. Karras, N. Zotos, A. Sideris, T. Moysiadis, A. Corsaro, G. Alexiou, C. Skianis, G. Mastorakis, C. X. Mavromoustakis, and E. Pallis, "EXEGESIS: Extreme edge resource harvesting for a virtualized fog environment," *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 173–179, Jul. 2017.

[36] P. G. V. Naranjo, Z. Pooranian, M. Shojafar, M. Conti, and R. Buyya, "Focan: A fog-supported smart city network architecture for management of applications in the Internet of everything environments," *J. Parallel Distrib. Comput.*, vol. 132, pp. 274–283, Oct. 2019.

[37] R. Mahmud, R. Kotagiri, and R. Buyya, "Fog computing: A taxonomy, survey and future directions," in *Internet Everything*. Cham, Switzerland: Springer, 2018, pp. 103–130.

[38] E. Baccarelli, P. G. V. Naranjo, M. Scarpiniti, M. Shojafar, and J. H. Abawajy, "Fog of everything: Energy-efficient networked computing architectures, research challenges, and a case study," *IEEE Access*, vol. 5, pp. 9882–9910, 2017.

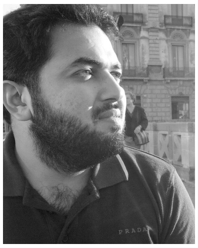
[39] L. F. Bittencourt, J. Diaz-Montes, R. Buyya, O. F. Rana, and M. Parashar, "Mobility-aware application scheduling in fog computing," *IEEE Cloud Comput.*, vol. 4, no. 2, pp. 26–35, Mar./Apr. 2017.

[40] S. Yi, Z. Hao, Z. Qin, and Q. Li, "Fog computing: Platform and applications," in *Proc. 3rd IEEE Workshop Hot Topics Web Syst. Technol. (HotWeb)*, Nov. 2015, pp. 73–78.

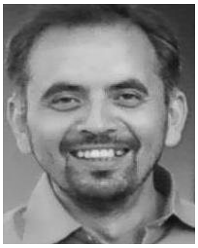
KAMRAN SATTAR AWAISI received the bachelor's degree from the COMSATS University Islamabad, Islamabad, Pakistan, where he is currently pursuing the master's degree, all in software engineering. His current research interest includes future Internet architectures, such as fog computing and software-defined networks.

ASSAD ABBAS received the Ph.D. degree in electrical and computer engineering from North Dakota State University, USA. He is currently an Assistant Professor in computer science with COMSATS University Islamabad, Islamabad, Pakistan. His current research interests include smart healthcare, big data analytics, fog computing, recommendation systems, patent analysis, software engineering, and social network analysis. Moreover, his research has appeared in several reputable international venues. He is also serving as the referee for numerous prestigious journals and as the technical program committee member for several conferences. He is a member of IEEE-HKN.

MAHDI ZAREEI (S'11–M'17) received the M.Sc. degree in computer network from the University of Science Malaysia, in 2011, and the Ph.D. degree from the Communication Systems and Networks Research Group, Malaysia–Japan International Institute of Technology, University of Technology Malaysia, in 2016. He is currently a Postdoctoral Fellow with the School of Engineering and Sciences, Tecnologico de Monterrey, Campus Monterrey. His current research interests include the IoT, cognitive radio, energy harvesting, 5G, and wireless sensor networks. He is a member of the Mexican National Researchers System, level I. He received the JASSO Scholarship to perform part of his Ph.D. research at Osaka University, Japan, in 2015. He is serving as an Associate Editor for IEEE ACCESS journal.



HASAN ALI KHATTAK (SM'19) received the Ph.D. degree in electrical and computer engineering from Politecnico di Bari, Bari, Italy, in 2015, the master's degree in information engineering from the Politecnico di Torino, Torino, Italy, in 2011, and the B.CS. degree in computer science from the University of Peshawar, Peshawar, Pakistan, in 2006. He has been an Assistant Professor in computer science with COMSATS University Islamabad, since 2016. His current research interests include web of things, data sciences, and social engineering for future smart cities. His perspective research areas include application of machine learning and data sciences for improving and enhancing quality of life in smart urban spaces through predictive analysis and visualization. He is a Professional Member of ACM and an Active Member of IEEE ComSoc, IEEE VTS, and the Internet Society.



MUHAMMAD USMAN SHAHID KHAN received the bachelor's degree in computer science from Bahauddin Zakariya University, Pakistan, in 2005, the master's degree in information security from the National University of Science and Technology (NUST), Pakistan, in 2008, and the Ph.D. degree in electrical and computer engineering from North Dakota State University, USA, in 2015. He is currently an Assistant Professor with COMSATS University Islamabad, Abbottabad Campus. His research interests include text analysis, data mining, social network analysis, recommendation systems, computer security, the IoT, and big data.

MAZHAR ALI received the M.S. degree from the Royal Institute of Technology (KTH), Stockholm, Sweden, in 2009, and the Ph.D. degree from the Department of Electrical and Computer Engineering, North Dakota State University (NDSU), Fargo, ND, USA, in 2015. He is currently an Assistant Professor with the COMSATS University Islamabad, Abbottabad, Pakistan. His current research interests include cloud computing, information security, smart health, and data and social network analysis.

IKRAM UD DIN (S'15–SM'18) received the M.Sc. degree in computer science and the M.S. degree in computer networking from the Department of Computer Science, University of Peshawar, Pakistan, and the Ph.D. degree in computer science from the School of Computing, Universiti Utara Malaysia (UUM). He was a member of the InterNetWorks Research Laboratory, UUM, from 2014 to 2016. He also served as the IEEE UUM Student Branch Professional Chair. He is currently a Lecturer with the Department of Information Technology, The University of Haripur, Haripur, Pakistan. He has 12 years of teaching and research experience in different universities/organizations. His current research interests include resource management and traffic control in wired and wireless networks, vehicular communications, mobility and cache management in information-centric networking, and the Internet of Things.

SAJID SHAH received the M.S. and Ph.D. degrees from Politecnico di Torino, Torino, Italy, in 2011 and 2015, respectively. He is currently an Assistant Professor with the Department of Computer Science, COMSATS University Islamabad, Abbottabad campus. His current research interests include data mining, text mining, machine learning, bioinformatics, and image processing. Moreover, he has been involved as a technical program committee member with many IEEE conferences and as a reviewer with several reputable journals.

• • •