

Article

# A Comprehensive and Effective Framework for Traffic Congestion Problem Based on the Integration of IoT and Data Analytics

Yazed Alsaawy <sup>1</sup>, Ahmad Alkhodre <sup>1,\*</sup>, Adnan Abi Sen <sup>1</sup>, Abdullah Alshantqi <sup>1</sup>, Wasim Ahmad Bhat <sup>1,2</sup> and Nour Mahmoud Bahbouh <sup>3</sup>

<sup>1</sup> Faculty of Computer and Information Systems, Islamic University of Madinah, Madinah 42351, Saudi Arabia; yalsaawy@iu.edu.sa (Y.A.); adnanmnm@iu.edu.sa (A.A.S.); amma@iu.edu.sa (A.A.); wab.cs@iu.edu.sa (W.A.B.)

<sup>2</sup> Department of Computer Sciences, University of Kashmir, Srinagar 190006, India

<sup>3</sup> Department of Computer Sciences, University of Granada, 18071 Granada, Spain; nourmahmoud@mail.ugr.es

\* Correspondence: aalkhodre@iu.edu.sa

**Abstract:** Traffic congestion is still a challenge faced by most countries of the world. However, it can be solved most effectively by integrating modern technologies such as Internet of Things (IoT), fog computing, cloud computing, data analytics, and so on, into a framework that exploits the strengths of these technologies to address specific problems faced in traffic management. Unfortunately, no such framework that addresses the reliability, flexibility, and efficiency issues of smart-traffic management exists. Therefore, this paper proposes a comprehensive framework to achieve a reliable, flexible, and efficient solution for the problem of traffic congestion. The proposed framework has four layers. The first layer, namely, the *sensing layer*, uses multiple data sources to ensure a reliable and accurate measurement of the traffic status of the streets, and forwards these data to the second layer. The second layer, namely, the *fog layer*, consumes these data to make efficient decisions and also forwards them to the third layer. The third layer, the *cloud layer*, permanently stores these data for analytics and knowledge discoveries. Finally, the fourth layer, the *services layer*, provides assistant services for traffic management. We also discuss the functional model of the framework and the technologies that can be used at each level of the model. We propose a smart-traffic light algorithm at level 1 for the efficient management of congestion at intersections, tweet-classification and image-processing algorithms at level 2 for reliable and accurate decision-making, and support services at level 4 of the functional model. We also evaluated the proposed smart-traffic light algorithm for its efficiency, and the tweet classification and image-processing algorithms for their accuracy.

**Keywords:** traffic; congestion; IoT; fog; cloud



**Citation:** Alsaawy, Y.; Alkhodre, A.; Abi Sen, A.; Alshantqi, A.; Bhat, W.A.; Bahbouh, N.M. A Comprehensive and Effective Framework for Traffic Congestion Problem Based on the Integration of IoT and Data Analytics. *Appl. Sci.* **2022**, *12*, 2043. <https://doi.org/10.3390/app12042043>

Academic Editors: Gianni Pantaleo, Agostino Forestiero and Pierfrancesco Bellini

Received: 1 December 2021

Accepted: 11 February 2022

Published: 16 February 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Thousands of people lose their lives annually due to road accidents, which also cause many disabilities and injuries, and contribute to other catastrophes through environment pollution due to traffic congestion [1]. Economic and health aspects are most significantly affected by traffic congestion due to delays in the execution of daily tasks, causing a huge waste of time, effort, and focus, as well as other additional costs [2]. Therefore, traffic congestion has become a fundamental problem and a major challenge for most countries of the world. Advances in digital technology in recent years have an effective role to play in solving this problem. As an example, the Internet of Things (IoT) has transformed everything around us into smart things [3], with a unique identity and the ability to sense and collect information from the surrounding environment, and share this information using radio-frequency identification (RFID) and wireless sensor networks (WSNs) [4]. RFID identifies an object (for example, a vehicle) using a unique identifier

that enables the surrounding environment to identify it, communicate with it, and track its location, whereas WSNs help the vehicle to obtain information such as images, sound, temperature, pressure, pollution level, and so on, from the surrounding environment. The analysis of these data can reveal a lot of useful information that can help to improve traffic congestion. The field of data science and its related technologies (such as machine learning, text mining, data mining, deep learning, and so on) can be leveraged for this purpose. However, the amount of data generated by these devices is huge and require high computing, storage, and energy capacity to process and analyze it [5,6]. Unfortunately, IoT devices have very limited computing resources at their disposal. Therefore, IoT devices usually resort to cooperating with the cloud for data storage and processing. However, due to inherent network latency problems, applications that demand real-time responses, such as healthcare, traffic monitoring, and so on, cannot be realized in this approach. Nevertheless, fog computing allows data from these sources to be collected by an initial station and enables users to apply policies to these data to filter, analyze, and discover any emergency events to deal with them immediately. Subsequently, data can be sent to the cloud for permanent storage, so that specific analyses can be conducted in the future to provide knowledge over a long period of time to facilitate decision-making. Furthermore, advances in data science, specifically through applications in other fields, has led to the emergence of many algorithms that can perform a variety of specialized tasks. In fact, these technologies are being used aggressively to provide smart and dynamic solutions in many areas of research. Unfortunately, this amalgam of technologies is yet to be investigated, applied, and tested in relation to solving the problem of traffic congestion.

In this paper, we propose a comprehensive and efficient framework for the traffic congestion problem, based on the integration of these modern technologies. The main contributions of this paper include:

1. A review of the literature related to traffic congestion.
2. A comprehensive framework for a reliable, flexible, and efficient solution for traffic congestion problems.
3. An algorithm for traffic lights for the efficient management of congestion at intersections and the evaluation of its efficiency.
4. An algorithm for the classification of tweets and the detection of congestion in a particular area and the evaluation of its accuracy.
5. An accuracy evaluation of an image-processing algorithm for congestion detection from images captured by drones.

The rest of this paper is organized as follows. Section 2 discusses the work related to smart traffic management, Section 3 presents the proposed framework, Section 4 describes a functional model of the framework, Section 5 discusses the experiments conducted and results obtained, and finally, Section 6 presents the conclusion of the study.

## 2. Related Work

This section discusses the approaches, techniques, technologies, and tools that have been suggested to overcome issues in traffic management. In the recent past, studies have been conducted to highlight many issues related to smart cities and these have emphasized the need for smart-traffic management [7–9]. To achieve this, many proposals have been made. In one such study, the use of data mining for classifying roads into six categories according to the traffic density (free, low, mid, high, very high, and extreme) was proposed [10]. Depending on the speed of vehicles and the average waiting time, this study predicted the expected congestion on a road and notified the end-user to select the appropriate route. Another study was able to predict traffic congestion by relating travel times with the vehicular load on the route [11]. Similarly, a study proposed a predictive road traffic management system based on the vehicular ad hoc network (VANET) architecture [12]. An algorithm to estimate the speed of a car from a video feed based on the image scale factor was also proposed [13]. More methods for addressing this issue using image processing can be found in [14]. Furthermore, techniques based on image processing

and deep learning for remote sensing images can be found in [15]. An overview of image processing algorithms and traffic management issues is presented in [16]. Applications of artificial intelligence (AI) in transportation systems have also been discussed in [17]. Ref. [18] studied the speed, acceleration, and direction of traffic flow using deep learning to manage the traffic flow.

In contrast, many studies have been conducted to investigate the role of sensors in traffic management. One study suggested deploying WSN and RFID on roads to create dynamic traffic lights [19]. However, the accuracy of these devices is affected by factors such as lighting, the random movement of vehicles, etc. Similar work, presented in [20], argued that with the help of a real-time intelligent central control unit, medical emergencies can be handled by pausing traffic in any lane to give priority to ambulances. This can be achieved by relying on RFID in cars and traffic lights. More such work on the application of sensors and RFID in smart traffic management can be found in [21]. A proposal to use scheduling algorithms to select the optimal path for an autonomous vehicle (AV) and suggestions to prepare special paths for this type of vehicle is discussed in [22]. Furthermore, an overview of AV is presented in [23]. However, the initial cost of new infrastructure is a major hurdle. The authors of one study [24] used micro-controllers and infrared (IR) sensors to detect traffic density and control the traffic signal to reduce traffic congestion and avoid unnecessary waiting times. A signal control project based on radio sensors with an Arduino low-power micro-controller has also been developed [25]. This controller works to capture and process information about traffic flow on roads in order to dynamically control the waiting time on the red light of a traffic signal. In a similar study, a method was presented of estimating the time of vehicles arriving at traffic lights based on their speed to inform drivers about street traffic status using computers in their cars [26].

Studies have also investigated the use of sensors for detecting accidents and traffic violations. One of the traditional ways to detect traffic violations is the use of surveillance cameras and radars. However, their accuracy is affected by weather conditions such as fog, heavy rain, and so on. Many researches have used modern technologies such as wireless devices, and drones to overcome these limitations of existing methods [27–30]. Generally, vehicles involved in an accident block the route, which leads to congestion. A European project employed a set of radio sensors to monitor the blocked and moving vehicles [31]. A study used Global Positioning System (GPS) technology to discover accidents by monitoring the speeds of vehicles to discover accidents and share this information with the accident control center, along with the location of the accident [16]. Another study made use of accelerometers for vehicles and GPS sensors in order to monitor the security and safety of commuters, and share the accident information with their relatives [32]. Ref. [17] employed smart devices in accident detection by implementing a special sensor for accident detection. Similarly, work described in [15] used the concept of dynamic time for accident detection through sensitive sensors in vehicles. A built-in circuit based on a micro-controller and sensors for the process of detecting the accident and then sending a signal about it has also been developed [33]. Another study [34] presented an accident control system that sends a text message after recognizing the occurrence of an accident through sensors that capture vibrations. The text message is sent to emergency services and contains the GPS location of the accident. Similar work can be found in [35]. Furthermore, other technologies, such as IoT [36], have been investigated for this purpose.

Monitoring and managing traffic using IoT has been proposed in the literature [37]. These studies generally rely on driver data to support decision-making in relation to traffic lights. It has been argued that as IoT connects physical things to the internet to build smart systems such as intelligent transportation systems (ITSs), communication among vehicles will create a new age of services in the land, sea, and air [38]. In fact, in the Internet of Vehicles (IoV), vehicles will act as sensing points, which bring more services, safety, and efficiency for transportation systems [39]. Moreover, the IoV can be used to provide sensors for environmental conditions, which can meet the needs of smart cities. It has also been suggested that integrating ITS with IoT can achieve the

goal of smart transportation [40]. The authors of that study also suggested the use of fog computing. In another such study [41], an increase in location-based service (LBS) applications, especially in smart cities with the IoT, has been observed. Based on this observation, the authors proposed a new framework for merging many IoT services in the transportation sector to monitor public transport. At the same time, researchers have also highlighted the need for a new level of security and privacy to protect the data of users in such a merger of technologies [42]. Preserving privacy is a vital issue in smart systems, which has to be considered because it is the main threat faced by the IoT and its applications [43,44]. Many studies have classified attackers that pose threats to the privacy of users' data and have discussed their skills [45]. Nevertheless, surveys on methods and techniques for preserving privacy have also been discussed [46,47]. A complete overview for the IoT and its structure, layers, phases, applications, and future trends can be found in [48,49].

Using big data to address the congestion issue has also been proposed [50]. Ref. [51] discussed the challenges and drivers of using machine learning algorithms with big data in the transportation field, using Hadoop and Map-Reduce to manage these data and to perform data analytics. The result can be useful in many applications such as parking, vehicle sharing, automatic vehicle positioning, etc. Using fog computing to address these challenges has also been suggested [52]. Ref. [53] discusses the challenges of using fog computing in ITSs.

On the other hand, smart transportation has extended beyond traffic management and is addressing vehicular transportation in general. Studies have investigated harvesting energy using roads, through solar energy and vibrational energy generated by vehicles. This energy is utilized for lighting either roads and traffic lights, and can be used to supplement the electricity grid [54]. Many countries with a large number of electric vehicles have undertaken projects to automatically charge electric vehicles using such technologies [55]. Another study makes use of musical songs generated by roads to warn commuters about the safety of their ride [56]. Many countries, such as Taiwan and Korea, are using this in a bid to avoid accidents related to unsafe driving. Furthermore, vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) technologies have been used to exchange critical information such as accident information between vehicles, and between vehicles and devices installed on the streets [57–59]. In this case, roads are equipped with electronic emergency services and vehicles are connected to VANETS [57–59]. Reducing the electricity consumption of street lights by controlling street luminance based on traffic flow has also been studied [54]. Studies have also developed techniques that enable measurements of the weight of static or mobile vehicles [54,60–62].

A summary of the literature reviewed in this study is presented in Table 1. It can be concluded that there is no single framework for addressing the reliability, flexibility, and efficiency issues of smart-traffic management. Therefore, a framework that acquires the traffic status of streets from multiple sources for reliability, supports the integration of different technologies for flexibility, and provides algorithms for efficiency is needed for smart-traffic management. This work intends to develop such a framework. We propose a comprehensive framework for smart-traffic management. The framework contains four layers. The first layer, namely, the *sensing layer*, uses multiple data sources to determine the reliability and accuracy of the traffic status of the streets, and forwards these data to the second layer. The second layer, the *fog layer*, consumes these data to make efficient decisions and also forwards them to the third layer. The third layer, the *cloud layer*, permanently stores these data for analytics and knowledge discovery. Finally, the fourth layer, the *services layer*, provides assistant services for traffic management. We also discuss the functional model of the framework and the technologies that can be used at each level of the model. We propose a smart-traffic light algorithm at level 1 for the efficient management of congestion at intersections, tweet-classification and image-processing algorithms at level 2 for reliable and accurate decision-making, and support services at level 4 of the functional model.

We also evaluated the proposed smart-traffic light algorithm for its efficiency, and tweet classification and image-processing algorithms for their accuracy.

**Table 1.** Summary of the literature reviewed in this study.

Literature	Central Idea of the Work(s)
[7,8]	Highlight many issues related to smart cities in which the need for smart-transportation has been emphasized
[10]	Use of data mining for classifying roads into six categories according to density
[13]	Estimates the speed of a car from a video based on the image scale factor
[14]	Methods for addressing traffic congestion using image processing
[15]	Techniques based on image processing and deep learning for remote sensing images
[16]	Overview of image processing algorithms and traffic issues
[17]	Applications of AI in transport systems
[18]	Deep learning to manage traffic flow
[19]	WSN and RFID on roads to create smart/dynamic traffic light
[20]	Real-time intelligent central control unit can deal with traffic emergencies
[21]	Application of sensors and RFID in smart traffic management
[22]	Uses scheduling algorithms to select the optimal path for an autonomous vehicle (AV)
[23]	An overview of AV is presented
[24]	Use micro-controllers and IR sensors to detect traffic density and control the traffic signal
[25]	A signal control project based on radio sensors with an Arduino low-power micro-controller
[26]	Estimating the time of vehicles arriving at traffic lights
[27–30]	Use modern technologies such as wireless devices, and drones to overcome limitations of surveillance cameras and radars
[31]	Employed a set of radio sensors to monitor blocked and moving vehicles
[32]	Use an accelerometer and GPS sensors to monitor the security and safety of commuters, and share the accident information with their relatives
[33]	Use a micro-controller and sensors for detecting accidents
[34–36]	Designed an accident control system that sends a text message after recognizing the accident
[37]	Monitoring and managing traffic using IoT
[38]	Argues that IoT connects physical things to the internet in order to build smart systems such as ITSs
[39]	Proposes using vehicles as sensing points to bring more services, safety, and efficiency for transport system
[40]	Integrating an intelligent transportation system (ITS) with IoT can achieve the goal of smart transportation
[41]	Found an increase in LBS applications, especially in smart cities with the IoT
[42]	Highlighted that a new level of security and privacy is required in the merging of IoT and smart transportation to protect users' data
[43,45]	Highlighted that preserving privacy is a vital issue in the smart systems
[46,47]	Surveys on methods and techniques for preserving privacy in smart transportation
[48,49]	A complete overview of IoT and its structure, layers, phases, applications, and future trends
[50]	Using big data to address the issue of traffic congestion
[51]	Discusses the challenges and drivers of using machine learning algorithms with big data in the transportation field
[52]	Uses fog computing to address smart transportation challenges
[53]	Discusses the challenges of using fog computing in ITS and highlights its contribution.
[54]	Harvesting energy generated on roads by vehicles for powering street and traffic lights
[55]	Automatically charging electric vehicles using energy generated by vehicles on roads
[56]	Makes use of musical songs generated by roads to warn commuters about the safety of their vehicle
[57–59]	Exchanges critical information such as accident information between vehicles, and between vehicles and devices installed in the streets
[60–62]	Develop techniques that enable the measurement of the weights of static or mobile vehicles

### 3. Proposed Framework

The proposed framework is based on four different layers that are integrated with each other to solve the problem. At each layer, several different solutions are offered in order to provide a high degree of reliability, efficiency, and flexibility. Figure 1 shows the general structure of the proposed framework.

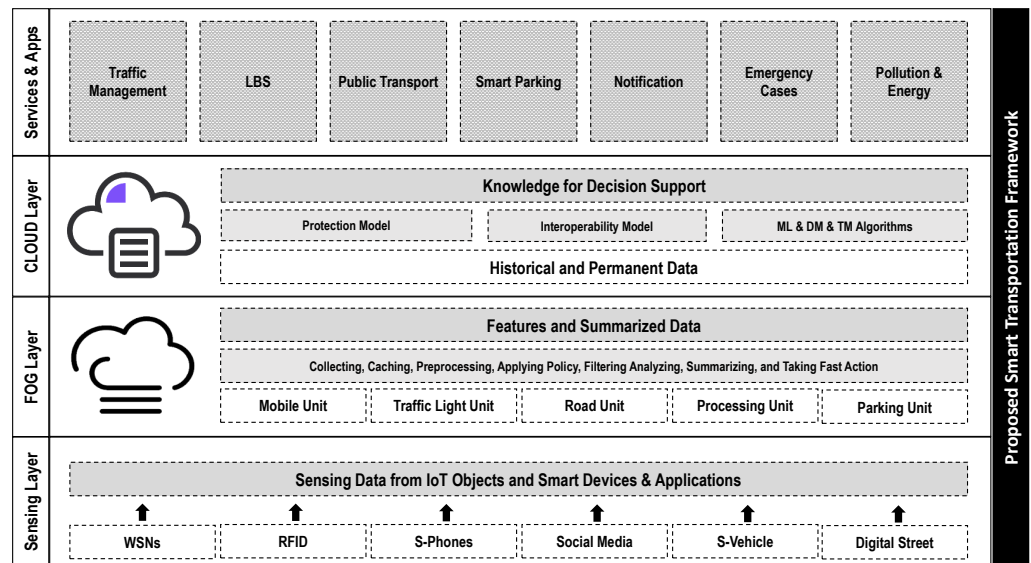


Figure 1. General structure of the proposed framework.

### 3.1. Sensing Layer

This layer collects data for the higher layers through various IoT infrastructures, such as wireless network sensors, radio identifiers, and so on. These infrastructures can be set up on streets and intersections, in addition to using devices embedded in vehicles and in the smartphones of drivers and passengers. As an example, social networking applications that are nowadays embedded in the dash panels of smart cars and are being used in many developed countries can be used as a source of data.

### 3.2. Fog Layer

This layer distributes its nodes densely across different places on main roads within cities and at intersections. These nodes can be fixed, such as a traffic lights or a side road unit. They may also be mobile, such as on a drone or a special vehicle that provides computing services. Each fog node is responsible for a small cell (or area) in which all the data generated from that area are collected, and the node processes those data, filters and reduces them, before sending them to the cloud. This is performed in order to reduce the amount of data to be transmitted, to eliminate useless data, and to thwart the inherent latency of the networks. At the same time, fog nodes provide fast computing services and process the received-data in real time. Consequently, they are able to respond in real time to any situation that requires an immediate response.

### 3.3. Cloud Layer

This layer collects and permanently stores all the information provided by the previous layer to generate a huge dataset based on continuous monitoring and tracking by the layers beneath it. Thus, these data can be relied on to train special models to discover the knowledge and behaviour of traffic, and the nature of each street or area across different times of the day, week, month, and year. This layer also has an important role to play in the process of managing the integration between heterogeneous applications, services and devices. Furthermore, this layer is responsible for maintaining the privacy and security of users' data.

### 3.4. Services/Applications Layer

This layer plays an important role in solving the traffic congestion problem as it can provide services such as pre-booking of car parking, supporting mass transportation, especially for students and employees, in addition to applications for medical emergencies, as well as applications that contribute to the sharing of information among users about

congestion in specific areas. Finally, various LBS applications, such as searching for points of interest or the shortest or fastest path, tracking, and other applications, can be realized in this layer. This layer enables service providers to add their services and their applications. The data generated by these applications and services will also serve as a source of new data in the cloud.

#### 4. Functional Model of the Framework

The functional model of the framework is presented in Figure 2. For each layer of the framework, there is a corresponding level in the functional model, highlighted in yellow in Figure 2. Our solution is based on dividing the traffic congestion problem into four levels, which include: (a) traffic lights at intersections, (b) main streets of the city, (c) city traffic-flow plan, and (d) assistant applications and services).

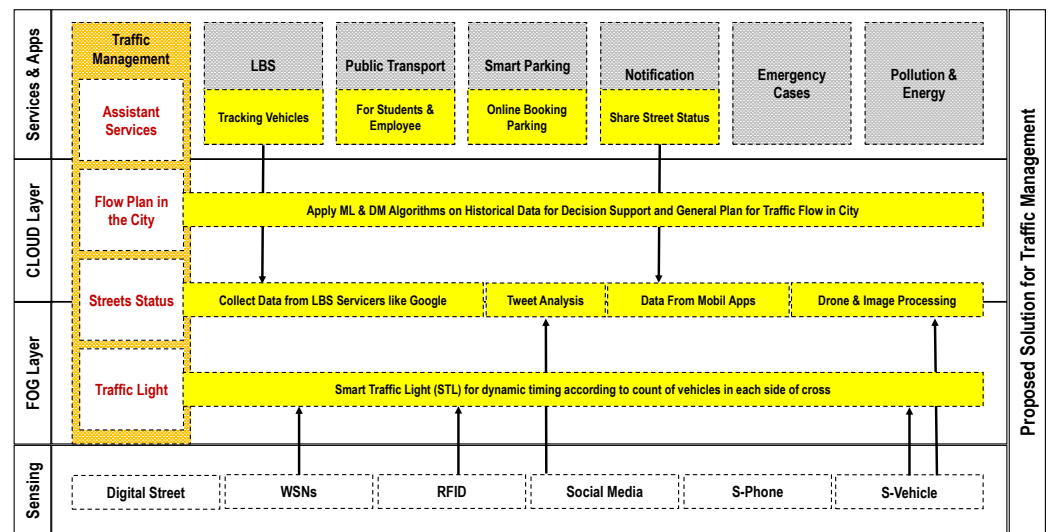


Figure 2. Functional model of the proposed framework with a description of each level.

To illustrate how these levels collaborate to solve traffic management issues, the following scenario is considered. If there is a traffic jam in a specific street, then the first level (smart traffic light) will give extra time to this street to ease the congestion. The second level will notice the condition of the street as congested and will send alerts to vehicles to stay away from this street currently and reroute their journey, which will further contribute to the speedy resolution of the problem. In this case, the third level is responsible for storing the data, and it will be noticeable during data analytics if this scenario occurs repeatedly. This will raise the alarm to solve this chronic problem by, for example, diverting the traffic at a specific time, or building a tunnel or bridge, and so on. Finally the fourth level, through various applications, such as the parking service, will help to alleviate the problem by restricting traffic to the congested area by providing stoppage places, and so on.

This framework is comprehensive because it is based on the principle of integrating many solutions together to create an effective solution. For example, if there is congestion on one side of a traffic signal, the smart-signal in the first level will increase the time of the green signal on the congested side to allow more vehicles to cross. However, at the same time, if the congestion is large and the waiting queue is long, the second level will notice that there is congestion at the street level and not only at the intersection. It will send alerts to drivers to avoid coming to this street and suggest them to take alternate routes to their destination. This will contribute significantly to resolving the congestion. The third level will store all the information according to the place and time, and after collecting huge volumes of these data, it is possible to discover the behavior of traffic on streets that are often crowded. Hence, this information will support decision-makers in taking action at the city level in re-planning the movement and flow of traffic by building

bridges or tunnels. Finally, at the fourth level, assistant applications contribute towards reducing the causes of congestion by facilitating tasks, such as looking for parking spaces in the city center (which otherwise cause very slow traffic movement) or using private vehicles for schools and work instead of mass transit (which otherwise cause crowding at peak times). In addition, as proposed in the framework, the diversity of data sources at each level, for the purpose of reliability, is also achieved in the functional model. In first level, we rely on cameras, WSNs, and RFIDs to know the number of vehicles in the street. In the second level, we rely on drones, tweet analysis, and a smartphone application that enables commuters to know the traffic status of the streets.

#### 4.1. Level 1—Traffic Congestion at Intersections

The problem of traffic congestion at intersections is the seeding level of a traffic jam, especially at peak times. The solution is to rely on intelligent, dynamic traffic signals adapted to the behaviour of traffic on each side of the intersection instead of static traffic lights that give a fixed time slot to traffic on each street. An alternative solution is building bridges or tunnels and this is always the best solution if available, but it is very expensive and may not be possible in all situations.

To create a smart traffic light, we need to know the number of vehicles in each street. We can acquire this information from three different types of sources. The first option is to rely on control cameras (either a camera in each street at the intersection or a moving camera that takes successive images). This is considered the easiest option in terms of its application, cost, and maintenance. This does not affect traffic during its installation or maintenance. However, the accuracy of the images may be influenced by luminance factors such as weather and lighting. Nevertheless, the threshold for tolerating the noise can be customized as we are interested in estimating congestion based on the size and number of vehicles. The second option is to rely on sensors on the roadside and the center of each road, separated by different distances, to measure the number of vehicles passing through. This ensures accuracy but may disturb traffic flow during installation and maintenance. The third option is to rely on assigning a unique radio ID to each vehicle, so that each vehicle connects to the nearest receiving unit (may be a traffic light). The location of each vehicle and the number of vehicles on each route can therefore be ascertained. Although this is effective, it may not be available in many countries for privacy reasons.

It is also possible to use more than one of these sources together to increase the level of availability and reliability, knowing that all the data will reach the fog node responsible for the intersection (which may be a traffic signal light) in order to process the data such as the images captured by cameras, calculate the congestion level, the number of vehicles on each route, and so on. Accordingly, the timing of the signals can be controlled so that a street with a huge number of waiting vehicles is given additional time at the expense of a side where there are no or few waiting vehicles. Indeed, starvation of each street is avoided by limiting the waiting time for any route to a specific amount. Moreover, all data pertaining to the congestion in each route and at each intersection are sent to the cloud for later use in the future planning of the city network and the flow of vehicles within it.

In our implementation of the framework, we used cameras at traffic lights, WSNs, and RFID to collect the data about traffic flow from the streets, i.e., the number of cars. This data are aggregated and fed to our proposed algorithm for a smart traffic light (STL), discussed below in Listing 1.



**Listing 1.** Proposed algorithm for a smart traffic light (STL).

```

1 Input: Repeat_Period = X // Scan period, say 10s
2 Output: Results R[ ] // Number of vehicles in each street
3 Initialization: Max_Time = 120; // Time allowed for green signal
4 Street1.WaitingTime=0;
5 Street2.WaitingTime=0;
6 Street3.WaitingTime=0;
7 Street4.WaitingTime=0;
8
9 Flag_MaxWait=0;
10
11 Start: Repeat execution each X seconds // say 10s
12 Step 1: Take new images of all streets
13 Step 2: For each street in Streets
14 R[Street] = Get_Objects_Num (street) // count the number of vehicles/
    street
15 If (Street.WaitingTime >= Max_Time)
16 Street.isGreen= True; // Open street
17 Street.WaitingTime = 0;
18 Flax_MaxWait=1;
19 Else
20 Street.WaitingTime = Street.waitingTime + X;
21 Flax_MaxWait=0;
22 End For;
23
24 Step 3: If (!Flag_MaxWait)
25 MaxStreet = Find_Max_Congestion ( R );
26 MaxSteet.isGreen = True; // Open street
27 MaxSteet.WaitingTime = 0;
28 END
29 END
30 Step 4: Go to Start

```

**4.2. Level 2—Traffic Congestion on Main Streets**

Congestion in main streets is usually due to obstacles such as excavations, traffic barriers, traffic accidents, rain and floods, and so on. Therefore, it is more important to discover the cause of congestion instead of just noting its occurrence. The solution is to then distribute the vehicular load of the congested street by informing the commuters about the condition of the street so that they can reroute to reach their destinations. Traffic lights can also be used to slow down the flow towards the congested street. The integration between different approaches listed below provides a higher level of accuracy and reliability for the information center to make such decisions.

1. Using information from service providers, such as Google Maps, OpenStreetMap, and so on, to determine the traffic status of streets and the degree of congestion, without specifying the cause or type of congestion. In our implementation of the framework, we used Google Maps.
2. Using crowd sourcing to exchange data between vehicles through a specific application in the event of any congestion or abnormal traffic on the street. This approach is more accurate as the incoming data are specific to the event, but it requires a high level of awareness from commuters in their use of the service. In future, this service can be automated within smart vehicles. In our implementation of the framework, we built an application for this service that works on smartphones, and contributes to the dissemination of this information to allow others to confirm the news.
3. Drones distributed at central vital points can be used to capture and share images to the information center to be processed automatically. This can provide information on the size of the congestion and help in distributing the vehicular load to other streets. In our implementation of the framework, we used an image processing algorithm to calculate the severity of the congestion that has been previously published [63].

4. Social media is one of the fastest ways to spread news today. In our implementation of the framework, we used Twitter. We proposed two algorithms based on text exploration and natural language processing to detect traffic congestion. The objective of first algorithm is to classify tweets related to the vehicular traffic in a particular street or area, whereas the second algorithm detects and reports the events related to congestion.

Listing 2 shows the algorithm used to classify the tweets, which is described below.

1. Collect tweets from Twitter.
2. Filter the collected tweets and extract only those that are related to the specific area.
3. Remove repeated tweets to reduce the total number of tweets.
4. Pre-process each tweet in the selected group.
  - (a) Clean the tweets of special letters, punctuation, letters from other languages, numbers, etc., and replace them with empty spaces.
  - (b) Encode the tweet to the list of symbols in each blank space.
  - (c) Remove the lexemes that do not affect the result of the classification (such as articles, pronouns, etc.) using the NLTK library [64].
  - (d) Normalize the text. This is needed when there are many similar characters with the same meaning.
  - (e) Stammering—return the word to its original root by removing some characters (such as the prefix or suffix) using Porter’s algorithm [64].

Note that the above listed steps reduce the size of the number of different words or terms by more than 60% compared to the original text.

5. Extract features and classify the tweets.
  - (a) One approach that can be used is based on machine learning to extract the features of each tweet by calculating the TF-IDF factor for each term. IDF refers to the number of tweets containing the term T divided by the total number of tweets. The algorithm uses  $IDF \cdot TF$ , where TF is calculated as shown in Equation (1)

$$TF(t) = \frac{\text{number of times } t \text{ appears}}{\text{Total number of terms}} \quad (1)$$

and IDF is calculated as shown in Equation (2)

$$IDF(t) = \ln\left(\frac{\text{Total number of docs}}{\text{Number of docs with term } t \text{ in them}}\right) \quad (2)$$

- (b) The second approach is based on the DMOZ dictionary [65]. This dictionary is used to calculate the number of words related to the vehicular traffic and if they are larger than a threshold value, the tweet is classified in that category.
- (c) The third approach is to find the POS for each term in the tweet. We can then measure the similarity with the root of a term for vehicular traffic and its branches in a WORDNET Tree in NLTK [64]. If the result is more than the minimum, the tweet is classified in that category.
- (d) Finally, an ontology of important words can be built based on previous traffic data so that the most important words that symbolize vehicular traffic are identified.

**Listing 2.** Proposed algorithm for the classification of tweets.

```

1 Tweets = Get_Tweets(); // Preprocessing phase
2 Terms = [] ; // Initial list for terms after preprocessing phase
3 Foreach Tweet in Tweets Do: // Cleaning
4 Tweet1= Tweet.Replace("Special Character" , " ");
5 // Repeat this line according to the Characters
6 // Tokenizing
7 Tweet_Words= Tweet1.Split(" ");
8 // Convert tweet to vector of words according to the space char
9 // Remove stop words
10 StopWordList= NLTK.Get_Stopwor(Lang=En);
11 // Prepare the list of English stop words based on NLTK
12 Foreach word in Tweet_Words Do:
13 If StopWordList.Contain (word) Then
14 Tweet_Words.Remove(Word Index)
15 // Remove
16 Else
17 //Stemmer
18 Term = NLTK.PorterStemmer( Tweet_Words(Word Index) );
19 Terms.Add (Term);
20 // create vector of terms after preprocessing phase
21 End~Foreach
22
23 // First Classification Phase
24 // There are three options
25 Class = ML_Method(Terms);
26 Class = DMOZ_Method(Terms);
27 Class = WORDNET_Method(Terms);
28
29 If Class = "Traffic" Then
30 //Start Second Classification Phase
31 Keywords = Get_Initial_Keywords_Of_Traffic();
32 //Expert will initialize this list
33 Count =0;
34 Foreach term in Terms Do:
35 If Keywords.Contain(term) Then
36 //matching
37 Count++;
38 End Foreach
39 Percentage_of_Matching = Count / Terms.Length();
40 If Percentage_of_Matching > Threshold Then
41 Add This Tweet to Sub_Class_of_Traffic (Class1++)
42 // example Class1 for Accident
43 End Foreach
44 If Class1.Count > Threshed2 Then
45 Send_Alert ()

```

Similarly, the algorithm shown in Listing 3 is used to identify tweets pertaining to vehicular traffic disruption, which is accomplished as follows:

1. Create a list of terms that symbolize a congestion or disruption of traffic flow in any street or area, such as an accident, police checkpoint, traffic stop, congestion, fire, heavy rain, repair work, and so on.
2. Check if the number of tweets containing one of the above listed terms is more than the minimum, which means that there is such an event in that area or street. Commuters in other areas driving to the event-area will be notified to avoid heading towards it, which will reduce the size of the problem, prevent it from getting worse, and speed up the process of resolving it.

**Listing 3.** Proposed algorithm for identifying tweets pertaining to a traffic disruption.

```

1 url = "Tweets_DataSet.txt"
2 Inputs = ['Feature1', 'Feature2', 'Class']
3
4 // Classify as Traffic or Not
5 // Feature1 is IDF, Feature2 is Tweet Length
6 dataset = pd.read_csv(url, names=Inputs) // Read
7 print(dataset.describe()) // Print Some Info about Data
8 array = dataset.values // Get All Data As~Table
9
10 //Split the Data to Features and Class
11 X = array[:,0:2] # All rows of cols (start from 0 to 2-1)
12 Y = array[:,2] # All rows of the col~3
13
14 // Split Data 80% for Training and 20% for Testing
15 x_train, x_test, y_train, y_test = model_selection.train_test_split(X, Y,
    test_size=0.2)
16
17 // Select the ML Algorithm
18 // K Nearest Neighbors, VSM, Random Forest, LG Regression etc.
19 model = KNeighborsClassifier()
20 model = SVM()
21 model = RandomForestClassifier(n_estimators=5)
22 model = LogisticRegression()
23
24 // Apply Training
25 model.fit(x_train,y_train)
26 predictions = model.predict(x_test)
27
28 // Apply Testing
29 print('KNN',accuracy_score(y_test, predictions))
30 // Select the Model of the Best Accuracy
31 // Call The Model for any new tweet after calculating Feature 1 and
    Feature 2
32 Res=model.predict([[F1,F2]]);
33 Return Res[0] // End Result

```

#### 4.3. Level 3—Analysis of Historical Data

This level performs the analysis of historical data pertaining to traffic flow to discover knowledge for city planning. Many technologies and applications have been employed in the first and second levels of the model. This implies that huge volumes of data are consequently collected in the cloud. These data provide a wealth of information that must be worked on to obtain the valuable information that will have a significant impact on improving the level of services and support in traffic management. The main objectives of this level are:

1. Applying machine learning algorithms to classify and aggregate data on congestion based on different criteria such as days, regions, and congestion level, as well as the cause of congestion. This enables the decision maker to develop correct strategies to avoid or resolve these issues.
2. Apply machine learning algorithms to classify and predict the percentage of congestion in a specific area at a specific time in the future and work in advance to avoid it.

It is worth mentioning here that this level will be implemented in future work when we have collected enough data for analysis over time. Nevertheless, this level has the following steps:

##### 4.3.1. First Step—Data Collection

The first step is to collect data coming from the applications and devices used in the first and second levels of the model, which mostly measure the level of congestion at a

specific time and place. We also rely on the accident database, which is mainly registered with government agencies.

The data received from WSNs are initially collected within the cache in the fog node and then sent periodically to the cloud to be stored in the database permanently in the format shown in Figure 3. The congestion can be represented by any one of the following values—normal, moderate, heavy, stop. The data received from smart traffic lights are stored in a similar format. However, the data from mobile applications are stored in the format shown in Figure 3. The reason field can have any one of the following values—students leaving, employees returning, barrier, weather conditions, excavations and maintenance, accident. The data received from the LBS, such as Google Maps, are shown in Figure 3. The data received as a result of tweet analysis are stored in the format shown in Figure 3. Finally, we merge all the data into one database to reduce the volume of data by removing duplicates based on the location, time and date, and so on. Moreover, the database is also used for storing the data pertaining to accidents.

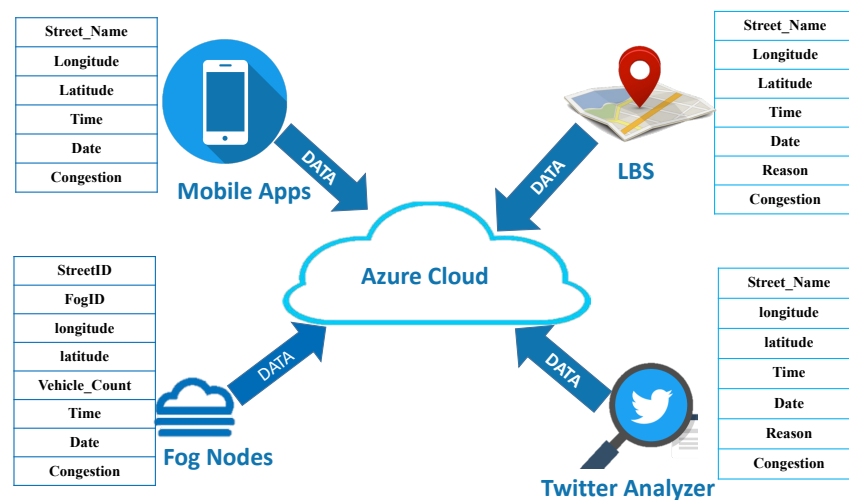


Figure 3. Schema for the data received by the cloud from different sources.

#### 4.3.2. Second Step—Data Processing and Cleaning

This step aims to process and clean the data for standardization and normalization. First, we process null values by either deleting the entire record or replacing the null with the mean or median of the values of other records. Second, outliers are also treated in the same way as null values. They are deleted or replaced with the mean or median value of other records in order to isolate their negative impact on the accuracy of the results and learning. Third, we convert the nominal or categorical data into numbers to enable their processing. As an example, the level of congestion is expressed in numerical values ranging from 1 to 4, with 4 being highest level of congestion. Subsequently, scaling is performed so that the value of each column is scaled between 0 and 1. Furthermore, we delete dummy data, using the One-Hot Encoder algorithm [64]. For example, after encoding the column for the congestion level, one of the previously mentioned four columns can be omitted because it is possible to deduce its value without its presence. In other words, if the value of the mean, the extreme, and the stop are equal to zero, then this means that the normal is one, without the need for it to exist in the database. Next, we process the data for standardization. For this purpose, we can use one of two methods: normalize or standardize, as shown in Equations (3) and (4).

$$N = \frac{[X - Min(x)]}{[Max(x) - Min(x)]} \tag{3}$$

$$S = \frac{[X - Mean(x)]}{SD(x)} \tag{4}$$

Furthermore, we merge or delete similar data to prevent duplication. Finally, we ensure that the data are balanced so that the value of each class of the output variable (e.g., the congestion level) is as close as possible to the number of records of the other class. Consequently, some records in small categories may be duplicated or some records in larger categories may be omitted.

#### 4.3.3. Third Step—Feature Selection

This step aims to determine the most important features of the collected and processed data. Several standard methods can be used to determine the most influential or important features, such as choosing all, backward elimination, forward selection, both backward and forward, all possible cases, or hiring an expert in the field.

#### 4.3.4. Fourth Step—Algorithm Selection

This step intends to use appropriate algorithms for classification and clustering. For classification, we use linear regression, logistic regression, decision tree, random forest, boost, naïve Bayes, support vector machine, K-nearest neighbor, neural network, and deep learning methods. For clustering, we use K-means, principal component analysis, and singular value decomposition methods [66].

#### 4.3.5. Fifth Step—Evaluation

In this step, we use various metrics for comparison and evaluation such as the confusion matrix, F-score, accuracy, performance, precision, recall, and so on [66]. After testing the results on new data, we evaluate whether the outperforming model suffers from overfitting or not, i.e., it is possible that the model was over-trained and therefore provided good results on the training data, and the results may change when the model is applied to the real dataset. However, if the model yields weak accuracy and did not train well, we may face a scenario of under-fitting. Therefore, we must reconsider the trained data, increase its size, or reconsider the selected features, and so on.

#### 4.3.6. Sixth Step—Visualization

The results are displayed graphically and in the form of different plots, depending on the requirements and services rendered.

### 4.4. Level 4—Support Services

We developed a set of useful applications that exploit the generated knowledge to reduce congestion. These applications include:

1. A parking reservation application that provides a pre-booking car-parking service and allows inquiries about the status of a parking space.
2. A driver awareness application that helps commuters to enhance their awareness about the traffic rules and the status of the streets.
3. An LBS-based application for searching points of interest.
4. An application that supports public transportation for schools, organizations, and government institutions to solve the problem of congestion.
5. An application that exploits smart signals to enable medical emergency vehicles such as ambulances to arrive their destination without any delays due to traffic congestion.
6. An application that exploits sensors to measure pollution and noise levels in cities to provide smarter services.

### 4.5. Technologies Used at Each Level

Figure 4 shows the technologies that were used at each level of the functional model.

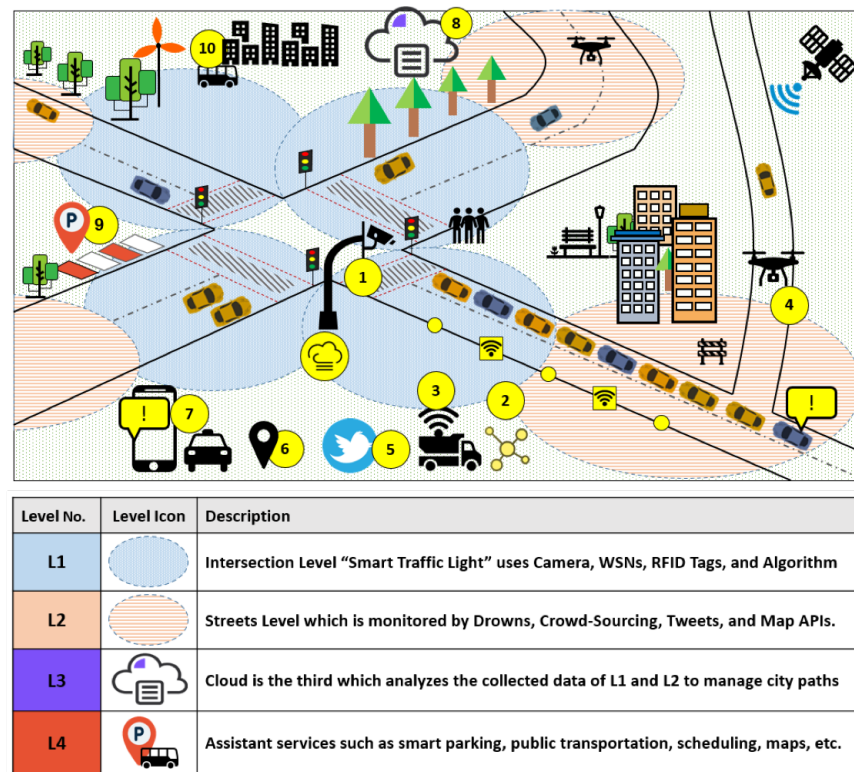


Figure 4. The levels at which technologies are applied in the functional model.

#### 4.5.1. First Level—Intersections

The proposed algorithm (see Listing 1) calculates the number of vehicles based on the data received from different sources. The algorithm decides the time of opening the signal for each street at the intersection based on the number of vehicles and the level of congestion in them. The technologies used include:

- A control camera (model number FoMaKo NDI PTZ Cameras): It captures an image of each side of the intersection periodically (every 10 s) and sends the image to the fog node to calculate the number of vehicles, shown as object 1 in Figure 4.
- WSNs are distributed on both sides of the road to calculate the number of cars passing. These sensors sense the presence of new cars once they pass through them and immediately send this information to the fog node shown as object 2 in Figure 4.
- Smart vehicles with embedded RFID tags are identified by RFID readers placed in the smart traffic lights, shown as object 3 in Figure 4.

We used a wireless network to link these devices. At the traffic light, the cameras send the images to fog nodes via a Wi-Fi network. The sensors and radio identifiers use two types of protocols (Bluetooth and Zigbee) to send their data to the side-unit or default gateway, which forwards that to a fog node via Wi-Fi. The fog node communicates with the cloud to send its data periodically using an internet connection.

#### 4.5.2. Second Level—Main Streets

The level of congestion discovered through several different methods on roads is used to alert commuters to choose other routes. These methods include:

- Drones (model number 4drc F11 Pro Com Câmera 4k Ptz) that capture and send images to the nearest station to be analyzed for the detection of any traffic congestion, shown as object 4 in Figure 4. The algorithm for processing these images has been published in our previous work [65].
- The proposed algorithm for analyzing tweets, shown as object 5 in Figure 4.

- APIs of service providers such as Google Maps or OpenStreetMap that return information about the traffic state of a specific location, shown as object 6 in Figure 4.
- A mobile application developed to enable users to collaborate by sending and confirming warnings about the traffic state on streets, shown as object 7 in Figure 4.

#### 4.5.3. Third Level—Cloud Level

The data from the first and second levels are collected for a period of time, and then analyzed to discover the behavior of traffic on roads at the city level, in order to support the decision-maker in taking special actions for city planning using Microsoft Azure.

#### 4.5.4. Fourth Level—Applications and Support Services

They have an indirect role in solving the congestion problem. The services provided include:

- A Smart Parking App to reserve parking in advance to avoid the search for stopping places, which otherwise results in congestion.
- The use of public transportation for school students or employees, instead of private cars, to avoid congestion at peak times. In addition, modifying the attendance times of workplaces and schools can contribute to reduce congestion.

### 5. Experiment and Results

In this section, we discuss the evaluation of the proposed framework. To do so, we evaluate each level of the functional model of the framework.

#### 5.1. Evaluation of Level 1

In this level, we proposed an algorithm to improve the efficiency of traffic lights at intersections. To evaluate our proposed algorithm for a smart traffic light (STL), we compare our work with ITMS [67]. ITMS also aims to improve the efficiency of a traffic light at an intersection by using differential time-slots for congested streets. Since STL also uses a similar approach, in our experiments we compared both algorithms (i.e., ITMS and STL) with a traditional traffic light (i.e., a fixed time-slot for each street). The main difference between STL and ITMS is that STL periodically (after every 10 s) checks the congestion rate on each street of the intersection, whereas ITMS checks only once and then opens the street with the maximum number of vehicles to pass all the vehicles. However, in both algorithms there is a maximum waiting limit for other signals that should not be exceeded.

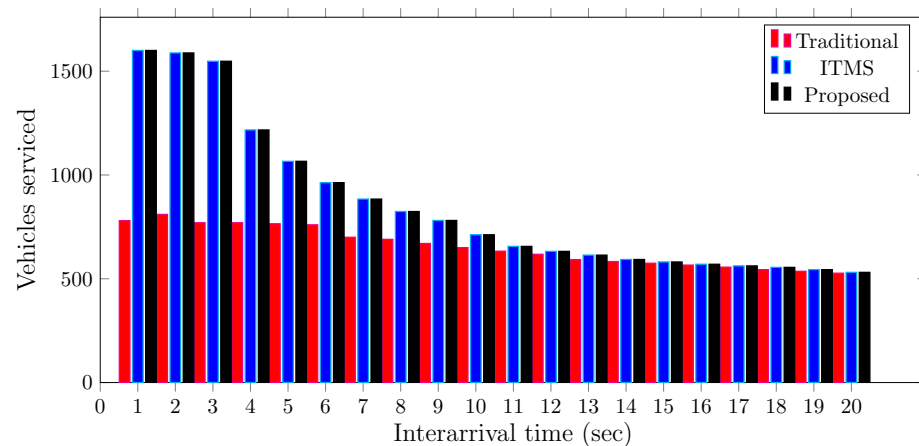
We used MATLAB for simulations to evaluate the algorithms (STL, ITMS and traditional) for the average waiting time and number of vehicles serviced for each street. We used same parameters during simulation as were used in the ITMS experiments. These were as follows:

- The total evaluation time was 1 h (3600 s).
- The number of streets at the intersection was 4.
- The inter-arrival time for vehicles on the first street ranged between 1 and 20 s per vehicle. This implies that the traffic congestion on the road decreases with the increase in this value.
- The inter-arrival time for vehicles on the second, third, and fourth streets was fixed at 30 s per vehicle. This implies that these streets were not congested.
- The time to get ready for an orange light was 2 s.
  - The maximum waiting time for any signal was 200 s.
  - The maximum opening time was 60 s (Green light).
  - The minimum opening time on the green light was 5 s (in case all cars had passed).
- The street without vehicles did not open at all.



### 5.1.1. Number of Serviced Vehicles

Figure 5 shows the results of our evaluation in terms of the number of vehicles serviced by STL, ITMS, and the traditional traffic light. When the inter-arrival time was small, congestion was more likely to happen. On the contrary, when the inter-arrival time was very large, congestion was less likely to happen. In Figure 5, it is evident that when the inter-arrival time was less than 10 s, the number of vehicles serviced by the proposed algorithm and ITMS was higher than that of the traditional method. In fact, for an inter-arrival time of 1 s, the number of vehicles serviced by the traditional method was less than half of that of the vehicles serviced by the proposed algorithm and ITMS. In the same figure, it is evident that as the inter-arrival time increased, the gap between them (STL/ITMS and traditional) in terms of the number of vehicles serviced became smaller. In fact, when the inter-arrival time reached around 20 s, all algorithms serviced the same number of vehicles. This implies that the proposed algorithm serviced a large number of vehicles compared to the traditional method in a high-congestion environment. ITMS also serviced a large number of vehicles compared to the traditional method in high-congestion scenarios. However, comparing ITMS and STL, the number of vehicles serviced by them was the same for all the inter-arrival times. This is expected in a scenario where there is only one street that is always congested. However, if there is no congestion, the performance of both systems tends to be close to the performance of the traditional one, as shown in the Figure 5.



**Figure 5.** Comparison of number of vehicles serviced by the traffic-light algorithms.

### 5.1.2. Average Waiting Time

Figure 6 shows the results of our evaluation of the average waiting time when one street is congested. It is evident from the figure that the average waiting time for the vehicles changed according to changes in the inter-arrival time of vehicles. STL and ITMS yielded similar results for all the inter-arrival times. Compared to the traditional traffic light, these algorithms yielded better results when the inter-arrival time was greater than 2 s and less than 8 s. This implies that when the street is highly congested, STL and ITMS significantly reduce the average waiting times of vehicles.

However, when two streets were congested, our experimental results demonstrate that STL performed better than ITMS when the inter-arrival time on both streets was less than 7 s, as shown in Figure 7. This implies that when two streets are highly congested, STL not only outperforms the traditional traffic light but also ITMS.

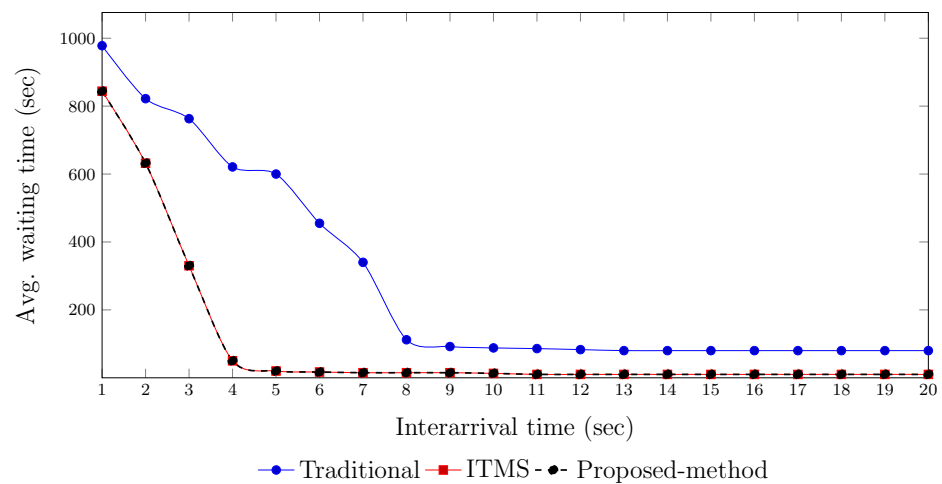


Figure 6. Comparison of average waiting times for the traffic-light algorithms—one street is congested.

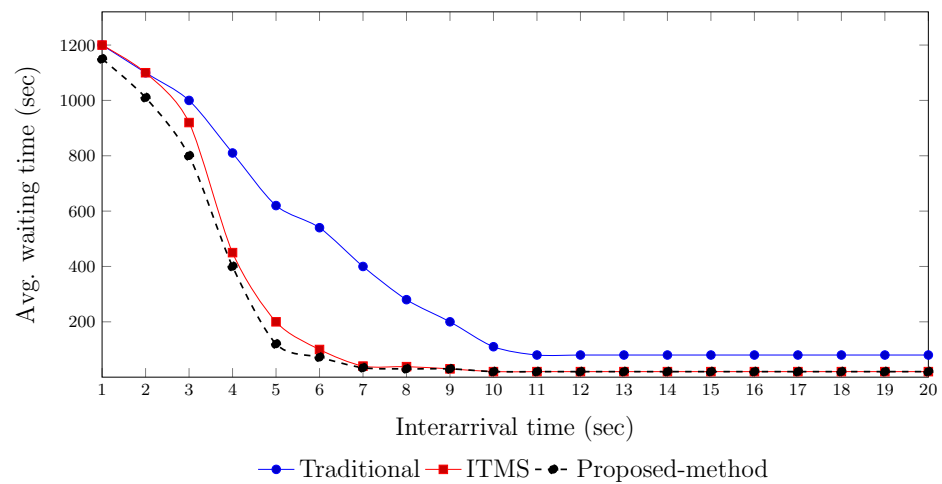


Figure 7. Comparison of average waiting times for the traffic-light algorithms—two streets are congested.

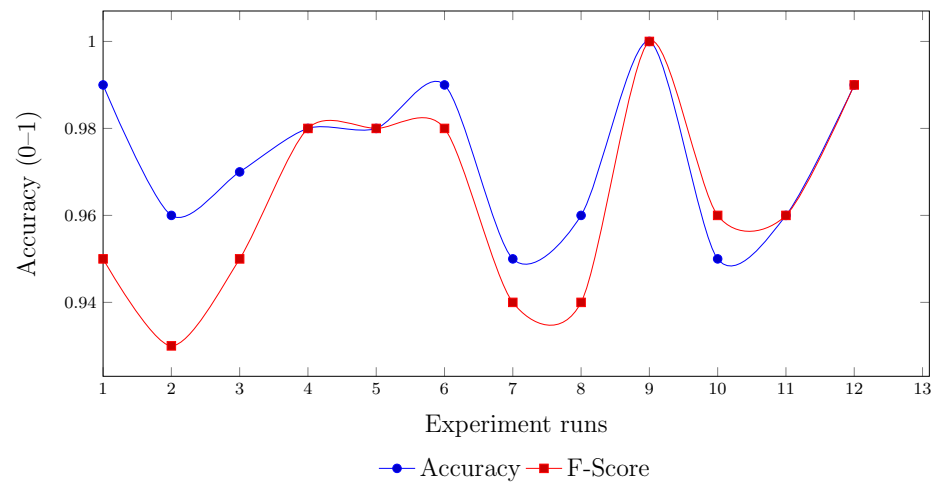
The results shown in Figures 5–7 clearly indicate that STL performed better than the traditional method in terms of the number of vehicles serviced and the average waiting times for vehicles when the streets were highly congested. These results also indicate that STL performed very closely to ITMS in terms of the number of vehicles serviced and the average waiting times of vehicles when a single street was highly congested. However, STL outperformed ITMS in reducing the average waiting times for vehicles when two streets were highly congested. This is because ITMS keeps the street open until all the cars have passed, which affects the congestion of the other street. In contrast, our algorithm distributes the time equally between the two streets. In other words, a more fair distribution of timing between the two streets takes place in our algorithm, which is the reason for its better performance in reducing the average waiting time.

### 5.2. Evaluation of Level-2

In this level, we proposed two natural language processing (NLP) algorithms to classify tweets based on their relevance to traffic congestion and detect the presence of traffic congestion from the identified tweets, with the possibility of determining the cause. We evaluated the accuracy of the proposed tweet classification and detection algorithms. We also evaluated the accuracy of the image processing algorithm proposed in our previous work [65].

### 5.2.1. Accuracy of Tweet Classification

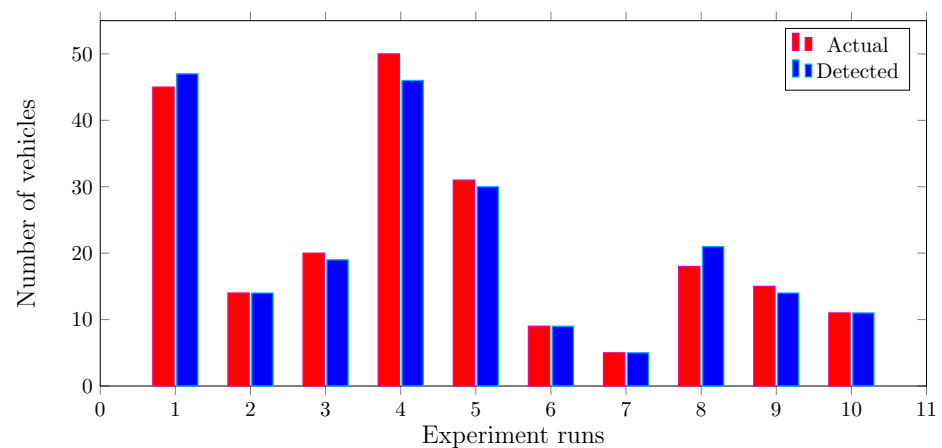
We implemented the proposed algorithm using the Python language and evaluated it on datasets collected from Twitter. Six datasets built using TF-IDF as a criterion for their importance were used to experiment with 12 queries (12 stories). Based on the number of important words belonging to the vehicular traffic in each tweet, TP, TN, FN, and FP were calculated. Finally, accuracy, precision, recall, and F-scores were calculated. The results shown in Figure 8 indicate that an average accuracy of nearly 97% was achieved by the proposed algorithm. This high accuracy is the result of a simple matching process with words belonging to the vehicular traffic.



**Figure 8.** Evaluation of the accuracy of tweet classification and detection of traffic congestion.

### 5.2.2. Accuracy of Images Processed

The accuracy of the processing of images captured by drones was proposed in our previous work [65]. We processed the images received from the drones using this algorithm to detect the approximate number of vehicles from any given image. Figure 9 shows the accuracy of the algorithm designed to detect the level of congestion from the images captured using drones. The results indicate that the number of vehicles detected by the algorithms was close to the actual number of vehicles in all the experiment runs, yielding an average accuracy of 95%.



**Figure 9.** Evaluation of the accuracy of the processing of images captured using drones for congestion detection.

### 5.3. Evaluation of Level-3

This level requires data to be collected over a period of time to train and use machine learning algorithms for the knowledge discovery process to facilitate city planning in relation to traffic flow. This study is part of a research project, and we are still in process of collecting data and designing algorithms for this level. Therefore, we were not yet able to evaluate this level in our experiments.

### 5.4. Validation of Level-4

In this level, we proposed a smartphone application to enable users to share information about the street status and receive alerts about traffic congestion on a street that is part of their route. Screenshots of the application are shown in Figure 10. We validated the working of the application in terms of its supported functionalities.

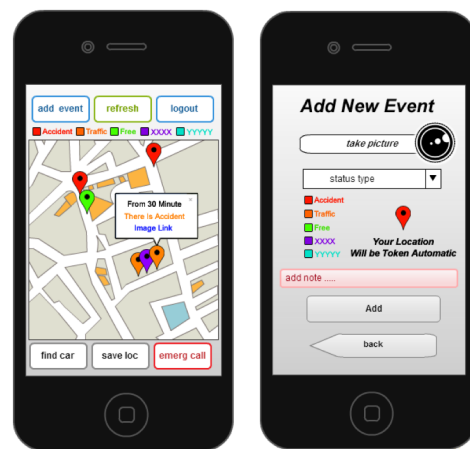


Figure 10. Screenshots of the application for sharing and receiving street traffic status.

## 6. Conclusions

In this paper, we presented a comprehensive framework that acquires the traffic-status of streets from multiple sources to ensure reliability, supports the integration of different technologies for flexibility, and provides algorithms for efficiency for the purpose of smart traffic management. We segregated and mitigated traffic congestion problems at different levels within the framework. We demonstrated that at each level various technologies can be used for addressing the issues relevant to that level. We proposed a smart-traffic-light algorithm in level 1 for the efficient management of congestion at intersections, tweet-classification and image-processing algorithms in level 2 for reliable and accurate decision support, and support services at level 4 of the functional model. We also evaluated the proposed smart-traffic-light algorithm in terms of its efficiency, and the tweet classification and image-processing algorithms for their accuracy. The results show that STL can minimize the average waiting times of vehicles at an intersection when two streets are crowded, and the use of social media and drones to detect traffic congestion can increase the reliability and accuracy of the STL. In our future work, we aim to implement and evaluate the third level of the framework. We also aim to investigate other challenges of the framework, such cost-benefit analyses, the safety and security of infrastructure, and the privacy of users' data.

**Author Contributions:** Conceptualization, Y.A., A.A. (Ahmad AlKhodre), A.A.S., A.A. (Abdullah Alshantqiti) and N.M.B.; methodology, A.A. (Ahmad AlKhodre), A.A.S., A.A. (Abdullah Alshantqiti) and N.M.B.; software, Y.A., A.A. (Ahmad AlKhodre) and A.A.S.; validation, A.A. (Ahmad AlKhodre) and A.A.S.; resources, A.A. (Abdullah Alshantqiti); data curation, N.M.B.; writing—original draft preparation, A.A. (Ahmad AlKhodre), A.A.S. and W.A.B.; writing—review and editing, A.A. (Ahmad AlKhodre), A.A.S. and W.A.B.; visualization, A.A. (Ahmad AlKhodre), A.A.S. and W.A.B.; supervision, A.A. (Ahmad AlKhodre) and A.A.S.; funding acquisition, Y.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Deanship of Scientific Research, Islamic University of Madinah, Saudi Arabia, under Tamayuz research grant number 2/710.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Isa, M.N.; Siyan, P. Analyzing factors responsible for road traffic accidents along Kano-Kaduna-Abuja Dual Carriageway Nigeria. *J. Econ. Sustain. Dev.* **2016**, *7*, 156–163.
2. Zadobrischi, E.; Cosovanu, L.M.; Dimian, M. Traffic Flow Density Model and Dynamic Traffic Congestion Model Simulation Based on Practice Case with Vehicle Network and System Traffic Intelligent Communication. *Symmetry* **2020**, *12*, 1172. [CrossRef]
3. Forestiero, A.; Papuzzo, G. Agents-based algorithm for a distributed information system in Internet of Things. *IEEE Internet Things J.* **2021**, *8*, 16548–16558. [CrossRef]
4. Greengard, S. *The Internet of Things*; MIT Press: Cambridge, MA, USA, 2021.
5. Bhat, W.A. Is a data-capacity gap inevitable in big data storage? *Computer* **2018**, *51*, 54–62. [CrossRef]
6. Bhat, W.A. Bridging data-capacity gap in big data storage. *Future Gener. Comput. Syst.* **2018**, *87*, 538–548. [CrossRef]
7. Alshekhy, I.F. Smart Cities: Survey. *J. Adv. Comput. Sci. Technol. Res.* **2012**, *2*, 79–90.
8. Misbahuddin, S.; Zubairi, J.A.; Saggaf, A.; Basuni, J.; Sulaiman, A.; Al-Sofi, A. IoT based dynamic road traffic management for smart cities. In Proceedings of the 2015 12th International Conference on High-Capacity Optical Networks and Enabling/Emerging Technologies (HONET), Islamabad, Pakistan, 21–23 December 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1–5.
9. Gul, F.; Mir, I.; Abualigah, L.; Sumari, P.; Forestiero, A. A Consolidated Review of Path Planning and Optimization Techniques: Technical Perspectives and Future Directions. *Electronics* **2021**, *10*, 2250. [CrossRef]
10. Jebamalar, J.P.A.; Paul, S.; Latha, D.P.P. Classifying Road Traffic Data Using Data Mining Classification Algorithms: A Comparative Study. In *Advances in Big Data and Cloud Computing*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 197–205.
11. Zambrano-Martinez, J.L.; Calafate, C.T.; Soler, D.; Lemus-Zúñiga, L.G.; Cano, J.C.; Manzoni, P.; Gayraud, T. A centralized route-management solution for autonomous vehicles in urban areas. *Electronics* **2019**, *8*, 722. [CrossRef]
12. Nafi, N.S.; Khan, R.H.; Khan, J.Y.; Gregory, M. A predictive road traffic management system based on vehicular ad-hoc network. In Proceedings of the 2014 Australasian Telecommunication Networks and Applications Conference (ATNAC), Southbank, VIC, Australia, 26–28 November 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 135–140.
13. Costa, L.R.; Rauen, M.S.; Fronza, A.B. Car speed estimation based on image scale factor. *Forensic Sci. Int.* **2020**, *310*, 110229. [CrossRef]
14. Hlehel, F.; Mansour, I.; Abu Saan, A. Video Based Car Speed Measurements and Detection. 2017. Available online: <https://repository.najah.edu/handle/20.500.11888/13148> (accessed on 1 February 2022).
15. Aloul, F.; Zualkernan, I.; Abu-Salma, R.; Al-Ali, H.; Al-Merri, M. iBump: Smartphone application to detect car accidents. *Comput. Electr. Eng.* **2015**, *43*, 66–75. [CrossRef]
16. Amin, M.S.; Jalil, J.; Reaz, M.B.I. Accident detection and reporting system using GPS, GPRS and GSM technology. In Proceedings of the 2012 International Conference on Informatics, Electronics & Vision (ICIEV), Dhaka, Bangladesh, 18–19 May 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 640–643.
17. Fanca, A.; Puscasiu, A.; Folea, S.; Vălean, H. Trauma accident detecting and reporting system. In Proceedings of the 2018 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR), Cluj-Napoca, Romania, 24–26 May 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–5.
18. Qu, L.; Li, W.; Li, W.; Ma, D.; Wang, Y. Daily long-term traffic flow forecasting based on a deep neural network. *Expert Syst. Appl.* **2019**, *121*, 304–312. [CrossRef]
19. Basavaraju, A.; Doddigarla, S.; Naidu, N.; Malgatti, S. Vehicle density sensor system to manage traffic. *IJRET Int. J. Res. Eng. Technol.* **2014**, *3*, 566–569.
20. Arifin, M.S.; Razi, S.A.; Haque, A.; Mohammad, N. A Microcontroller Based Intelligent Traffic Control System. *Am. J. Embed. Syst. Appl.* **2019**, *7*, 21–25.
21. Guerrero-Ibáñez, J.; Zeadally, S.; Contreras-Castillo, J. Sensor technologies for intelligent transportation systems. *Sensors* **2018**, *18*, 1212. [CrossRef]
22. Lam, A.Y.; Leung, Y.W.; Chu, X. Autonomous-vehicle public transportation system: Scheduling and admission control. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 1210–1226. [CrossRef]

23. Tokody, D.; Mezei, I.J.; Schuster, G. An overview of autonomous intelligent vehicle systems. In *Vehicle and Automotive Engineering*; Springer: Cham, Switzerland, 2017; pp. 287–307.
24. Ghazal, B.; Elkhatib, K.; Chahine, K.; Kherfan, M. Smart traffic light control system. In Proceedings of the 2016 Third International Conference on Electrical, Electronics, Computer Engineering and Their Applications (EECEA), Beirut, Lebanon, 21–23 April 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 140–145.
25. Jagadeesh, Y.; Suba, G.M.; Karthik, S.; Yokesh, K. Smart autonomous traffic light switching by traffic density measurement through sensors. In Proceedings of the 2015 International Conference on Computers, Communications, and Systems (ICCCS), Kanyakumari, India, 2–3 November 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 123–126.
26. Möller, D.P.; Fidencio, A.X.; Cota, E.; Jehle, I.A.; Vakilzadian, H. Cyber-physical smart traffic light system. In Proceedings of the 2015 IEEE International Conference on Electro/Information Technology (EIT), Dekalb, IL, USA, 21–23 May 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 546–551.
27. Toh, C.K.; Cano, J.C.; Fernandez-Laguia, C.; Manzoni, P.; Calafate, C.T. Wireless digital traffic signs of the future. *IET Netw.* **2019**, *8*, 74–78. [[CrossRef](#)]
28. Toh, C.K. Future application scenarios for MANET-based intelligent transportation systems. In Proceedings of the Future Generation Communication and Networking (FGCN 2007), Jeju, Korea, 6–8 December 2007; IEEE: Piscataway, NJ, USA, 2007; Volume 2, pp. 414–417.
29. Englund, C.; Chen, L.; Vinel, A.; Lin, S.Y. Future applications of VANETs. In *Vehicular ad hoc Networks*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 525–544.
30. Wang, L.; Chen, F.; Yin, H. Detecting and tracking vehicles in traffic by unmanned aerial vehicles. *Autom. Constr.* **2016**, *72*, 294–308. [[CrossRef](#)]
31. Schirokoff, A.; Pilli-Sihvola, E.; Sihvola, N. Assessing the safety impacts of intersection safety systems. *Procedia-Soc. Behav. Sci.* **2012**, *48*, 1515–1524. [[CrossRef](#)]
32. Chikaka, T.P.; Longe, O.M. An Automatic Vehicle Accident Detection and Rescue System. In Proceedings of the 2021 IEEE 6th International Forum on Research and Technology for Society and Industry (RTSI), Naples, Italy, 6–9 September 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 418–423.
33. Megalingam, R.K.; Nair, R.N.; Prakhya, S.M. Wireless vehicular accident detection and reporting system. In Proceedings of the 2010 International Conference on Mechanical and Electrical Technology, Singapore, 10–12 September 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 636–640.
34. Ahire, H.S.; Kamble, M.B.; Khade, P.K.; Ghare, R.A.; Jadhav, B. Vehicle accident detection and alerting system. *Int. J. Res. Sci. Eng. Technol.* **2018**, *6*, 67. [[CrossRef](#)]
35. Parmar, P.; Sapkal, A.M. Real time detection and reporting of vehicle collision. In Proceedings of the 2017 International Conference on Trends in Electronics and Informatics (ICEI), Tirunelveli, India, 11–12 May 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1029–1034.
36. Nath, I.; Dey, A.; Das, S.; Battavayal, R.; Ballav Garai, B.; Majumder, A. A New Accident Alert System Using Android App Development. In Proceedings of the 2nd International Conference on Non-Conventional Energy: Nanotechnology & Nanomaterials for Energy & Environment (ICNNEE), Kolyani, India, 18–19 October 2019.
37. Parekh, S.; Dhami, N.; Patel, S.; Undavia, J. Traffic signal automation through IoT by sensing and detecting traffic intensity through IR sensors. In *Information and Communication Technology for Intelligent Systems*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 53–65.
38. Muthuramalingam, S.; Bharathi, A.; Gayathri, N.; Sathiyaraj, R.; Balamurugan, B. IoT based intelligent transportation system (IoT-ITS) for global perspective: A case study. In *Internet of Things and Big Data Analytics for Smart Generation*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 279–300.
39. Ang, L.M.; Seng, K.P.; Ijamaru, G.K.; Zungeru, A.M. Deployment of IoV for smart cities: Applications, architecture, and challenges. *IEEE Access* **2018**, *7*, 6473–6492. [[CrossRef](#)]
40. Hussain, M.M.; Khan, F.; Alam, M.S.; Beg, M.S. Fog computing for ubiquitous transportation applications—A smart parking case study. In *Engineering Vibration, Communication and Information Processing*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 241–252.
41. Mostafa, M.H. Internet of Things-Based Framework for Public Transportation Fleet Management in Non-smart City. In Proceedings of the Emerging Technologies for Developing Countries: Second EAI International Conference, AFRICATEK 2018, Cotonou, Benin, 29–30 May 2018; Springer: Berlin/Heidelberg, Germany, 2018; Volume 260, p. 3.
42. Ma, L.; Huo, Y.; Hu, C.; Li, W. Security and Privacy for Smart Cyber-Physical Systems. *Secur. Commun. Netw.* **2019**, *2019*, 7045862. [[CrossRef](#)]
43. Abi Sen, A.A.; Eassa, F.A.; Jambi, K. Preserving privacy of smart cities based on the fog computing. In Proceedings of the International Conference on Smart Cities, Infrastructure, Technologies and Applications, Jeddah, Saudi Arabia, 27–29 November 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 185–191.
44. Forestiero, A. Metaheuristic algorithm for anomaly detection in Internet of Things leveraging on a neural-driven multiagent system. *Knowl.-Based Syst.* **2021**, *228*, 107241. [[CrossRef](#)]
45. Shin, K.G.; Ju, X.; Chen, Z.; Hu, X. Privacy protection for users of location-based services. *IEEE Wirel. Commun.* **2012**, *19*, 30–39. [[CrossRef](#)]

46. Wernke, M.; Skvortsov, P.; Dürr, F.; Rothermel, K. A classification of location privacy attacks and approaches. *Pers. Ubiquitous Comput.* **2014**, *18*, 163–175. [[CrossRef](#)]
47. Abi Sen, A.A.; Eassa, F.A.; Jambi, K.; Yamin, M. Preserving privacy in internet of things: A survey. *Int. J. Inf. Technol.* **2018**, *10*, 189–200. [[CrossRef](#)]
48. Lin, J.; Yu, W.; Zhang, N.; Yang, X.; Zhang, H.; Zhao, W. A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications. *IEEE Internet Things J.* **2017**, *4*, 1125–1142. [[CrossRef](#)]
49. Atzori, L.; Iera, A.; Morabito, G. The internet of things: A survey. *Comput. Netw.* **2010**, *54*, 2787–2805. [[CrossRef](#)]
50. Rathore, M.M.; Paul, A.; Hong, W.H.; Seo, H.; Awan, I.; Saeed, S. Exploiting IoT and big data analytics: Defining smart digital city using real-time urban data. *Sustain. Cities Soc.* **2018**, *40*, 600–610. [[CrossRef](#)]
51. Babar, M.; Arif, F.; Jan, M.A.; Tan, Z.; Khan, F. Urban data management system: Towards Big Data analytics for Internet of Things based smart urban environment using customized Hadoop. *Future Gener. Comput. Syst.* **2019**, *96*, 398–409. [[CrossRef](#)]
52. Darwish, T.S.; Bakar, K.A. Fog based intelligent transportation big data analytics in the internet of vehicles environment: Motivations, architecture, challenges, and critical issues. *IEEE Access* **2018**, *6*, 15679–15701. [[CrossRef](#)]
53. Hussain, M.; Alam, M.S.; Beg, M. Fog computing in IoT aided smart grid transition-requirements, prospects, status quos and challenges. *arXiv* **2018**, arXiv:1802.01818.
54. Toh, C.K.; Sanguesa, J.A.; Cano, J.C.; Martinez, F.J. Advances in smart roads for future smart cities. *Proc. R. Soc. A* **2020**, *476*, 20190439. [[CrossRef](#)] [[PubMed](#)]
55. Boffey, D. World's first electrified road for charging vehicles opens in Sweden. *The Guardian*, 13 April 2018.
56. Zhou, M.; Huang, D.; Hu, Y.; Zhou, L.; An, L. Musical roads: Design, construction and potential economic and safety benefits. In *Proceedings of the Institution of Civil Engineers-Transport*; Thomas Telford Ltd.: London, UK, 2018; pp. 1–27. [[CrossRef](#)]
57. IEC, I. *Smart Cities Preliminary Report*; ISO/IEC JTC1: Geneva, Switzerland, 2014.
58. Martinez, F.J.; Toh, C.K.; Cano, J.C.; Calafate, C.T.; Manzoni, P. Emergency services in future intelligent transportation systems based on vehicular communication networks. *IEEE Intell. Transp. Syst. Mag.* **2010**, *2*, 6–20. [[CrossRef](#)]
59. Umedu, T.; Isu, K.; Higashino, T.; Toh, C.K. An intervehicular-communication protocol for distributed detection of dangerous vehicles. *IEEE Trans. Veh. Technol.* **2009**, *59*, 627–637. [[CrossRef](#)]
60. Jacob, B.; Feypell-de La Beaumelle, V. Improving truck safety: Potential of weigh-in-motion technology. *IATSS Res.* **2010**, *34*, 9–15. [[CrossRef](#)]
61. Winter, K.; Campbell, B. *Technology Helps Us Do Things Better: Virtual Weigh Stations, Other New Methods for Enforcing the Law Prolong Highway Life and Increase Vehicle Citations*; Virginia Department of Transportation: Richmond, VA, USA, 2009.
62. Jacob, B.; O'Brien, E.; Jehaes, S. *Weigh-in-Motion of Road Vehicles: Final Report of the COST 323 Action*; LCPC: Paris, France, 2002.
63. Abi Sen, A.A.; Alsaawy, Y.; Alkhodre, A.; Bahbouh, N.M.; Alhaboob, M. Smart Street Design for Managing Crowds in Hajj. In *Proceedings of the 2021 8th International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, India, 17–19 March 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 385–389.
64. Roul, R.K.; Gugnani, S.; Kalpeshbhai, S.M. Clustering based feature selection using extreme learning machines for text classification. In *Proceedings of the 2015 Annual IEEE India Conference (INDICON)*, New Delhi, India, 17–20 December 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1–6.
65. Alharbi, A.; Halikias, G.; Sen, A.A.A.; Yamin, M. A framework for dynamic smart traffic light management system. *Int. J. Inf. Technol.* **2021**, *13*, 1769–1776. [[CrossRef](#)]
66. Bokaba, T.; Doorsamy, W.; Paul, B.S. Comparative study of machine learning classifiers for modelling road traffic accidents. *Appl. Sci.* **2022**, *12*, 828. [[CrossRef](#)]
67. Elsaygher Mohamed, S.A.; AlShalfan, K.A. Intelligent Traffic Management System Based on the Internet of Vehicles (IoV). *J. Adv. Transp.* **2021**, *2021*, 4037533. [[CrossRef](#)]