Contents lists available at ScienceDirect

Egyptian Informatics Journal

journal homepage: www.sciencedirect.com

Full length article

Smart cities: Fusion-based intelligent traffic congestion control system for vehicular networks using machine learning techniques

Muhammad Saleem [a], Sagheer Abbas [a], Taher M. Ghazal [b,c], Muhammad Adnan Khan [d,e,*], Nizar Sahawneh [f], Munir Ahmad [a]

[a] School of Computer Science, National College of Business Administration and Economics, Lahore 54000, Pakistan
[b] Center for Cyber Security, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia (UKM), 43600 Bangi, Selangor, Malaysia
[c] School of Information Technology, Skyline University College, University City Sharjah, 1797 Sharjah, UAE
[d] Pattern Recognition and Machine Learning Lab, Department of Software, Gachon University, Seongnam 13557, South Korea
[e] Riphah School of Computing & Innovation, Faculty of Computing, Riphah International University, Lahore Campus, Lahore 54000, Pakistan
[f] School of Business, Skyline University College, University City Sharjah, 1797 Sharjah, UAE

ARTICLE INFO

Article history:
Received 12 February 2022
Revised 11 March 2022
Accepted 17 March 2022
Available online 13 April 2022

Keywords:
Vehicular networks
Smart city
Machine learning
Fusion

ABSTRACT

Smart cities have been developed over the past decade, and reducing traffic congestion has been the top concern in smart city development. Short delays in communication between vehicles and Roadside Units (RSUs), smooth traffic flow, and road safety are the key challenges of Intelligent Transportation Systems (ITSs). The rapid upsurge in the number of road vehicles has increased traffic congestion and the number of road accidents. To fix this issue, Vehicular Networks (VNs) have developed many new ideas, including vehicular communications, navigation, and traffic control. Machine Learning (ML) is an efficient approach to finding hidden insights into ITS without being programmed explicitly by learning from data. This research proposed a fusion-based intelligent traffic congestion control system for VNs (FITCCS-VN) using ML techniques that collect traffic data and route traffic on available routes to alleviate traffic congestion in smart cities. The proposed system provides innovative services to the drivers that enable a view of traffic flow and the volume of vehicles available on the road remotely, intending to avoid traffic jams. The proposed model improves traffic flow and decreases congestion. The proposed system provides an accuracy of 95% and a miss rate of 5%, which is better than previous approaches.

1. Introduction

Smart city development is coupled with a significant shift regarding planning and adopting advanced technologies to assemble "smarter" cities to enhance people's quality of life. The European Commission launched innovative and apparent creativity on smart cities in four areas: electricity, buildings, heating and cooling systems, and transport [1]. An intelligent transportation solution can improve traffic flow in smart cities by monitoring traffic patterns and adjusting traffic signal timing. The aim is to ascertain and assist sustainable forms of transportation, to boost an Intelligent Transportation System (ITS) occupying real-time information, Traffic Management Systems (TMSs) to avoid congestion, safety, and green applications (e.g. to minimize the utilization of fuel, gas, and energy) [2]. ITS leverages novel and emerging technologies to make mobility more pleasant and cost-effective in a smart city.

In recent years, one of the fundamental dilemmas with regard to transportation systems has been traffic congestion, which must be solved to minimize fuel waste, accidents, traffic jams, and driver frustration. The majority of traffic delays in metropolitan areas occur because of the high number of vehicles. Traffic regulation

Abbreviations: ITS, Intelligent Transportation System; TMSs, Traffic Management Systems; ML, Machine Learning.

* Corresponding author at: Pattern Recognition and Machine Learning Lab, Department of Software, Gachon University, Seongnam 13557, South Korea.

E-mail addresses: adnan@gachon.ac.kr, adnan.khan@riphah.edu.pk (M. Adnan Khan).

Peer review under responsibility of Faculty of Computers and Information, Cairo University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.eij.2022.03.003

during rush hours is an important issue [3]. Transportation networks are becoming an integral part of human life due to the shortage of land resources and saturated transportation infrastructure in metropolitan areas. As a result of this overcrowding, numerous traffic-related drawbacks have arisen in urban zones where people must move quickly from one location to another [4].

With the recent advances in Artificial Intelligence (AI), and Machine Learning (ML) ITS, and smart environmental monitoring in smart cities monitor the factors influencing the environment more precisely, with an optimum control of pollution, traffic congestion and other adverse effects. Traffic congestion affects people's quality of life because it reduces traffic performance and increases severe environmental pollution. Hence, the country's production, economic development, and human activities are influenced by traffic congestion. The most pressing issue in urban planning is determining how to address traffic congestion effectively [5]. Traffic congestion management is a significant area of study, with many solutions emerging from miscellaneous research projects in the field over the last several decades [6]. As time passes, traffic data collection and intelligent transportation systems have evolved in response to these issues [7].

Traffic congestion management is critical for making it easier to drive on highways. Most traffic signals are preprogrammed. This does not support real-time conditions and causes traffic congestion. There is an increased need for density-driven traffic movement that is based on existing traffic conditions. Traffic junction signals play a crucial role in reducing congestion [8]. Another reason for traffic congestion is a lack of road infrastructure. The driver's waiting time has risen as a result. This is primarily due to the inefficient operation of traffic signals. Approaches focused on Vehicle-to-Vehicle (V2V) communication are unable to reliably measure traffic congestion volume. Additionally, traffic signalling systems with a fixed operating period cannot handle shifting traffic volumes, resulting in long traffic queues at road crossings [9].

ITS includes intelligent traffic signal controls, highway management, and emergency services management. These systems capture real-time traffic data and proceed with required steps to prevent road traffic congestion. Traffic congestion on the roads of cities can be predicted precisely by smartphone applications such as Google and Apple Maps based on sensor data collected from highways and city road monitoring devices [10]. Real-time traffic alerts on exit roads may help the driver choose the best route from his present location. Drivers like to be informed about the state of congestion at upcoming intersections in order to save time.

The proposed system is based on VNs coupled with fused ML techniques, measures the density of traffic congestion, and provides the user with an alternative path for travelling by using a mobile application that can save time.

2. Literature review

In the past several years [2–7], a number of studies have focused on the topic of traffic congestion. Using Information and Communication Technology (ICT) [11] and Internet of Things (IoT) [12,13] applications, multiple types of research to monitor road traffic congestion and traffic control have been suggested in the literature to increase the efficiency of the current TCCS. Using a real-time dynamic traffic control scheme to route vehicles smartly will improve traffic movement on roads, allowing facilities to be used more effectively and creating an atmosphere conducive to resolving city gridlock.

The Internet of Vehicles (IoV) is a network of vehicles equipped with sensors that may play a prime role in TMS by linking physical devices over the internet to provide more precise, quick, and accurate results. In IoV, every database is stored on a computer via the internet. Remote access to IoV components reduces human involvement [14].

VN is a specific case of wireless communication technology that is hampered by rapid topological changes due to the fast movement of vehicle nodes [15]. Given the rising quantity of vehicles prepared by wireless and communication devices, intravehicular communication has become an auspicious field of research. VNs empower many applications by providing wireless communication technology to vehicle nodes. This can involve accident prevention, dynamic route planning, and real-time traffic status monitoring. VN is a new mobile network (MN) that includes vehicles that manage themselves as mobile nodes. VN has been proposed to enhance safety and comfort with the aid of V2V and Vehicle-To-Roadside (V2R) communications. VNs have been created to improve driver safety and comfort with the assistance of V2V and V2R communications in vehicular environments [16,17]. VNs are now considered an infrastructure for ITSs with the growing number of autonomous vehicles in smart cities. The deployment of a VN is a solution to communicate between vehicles [18]. The architecture of a VN is presented in Fig. 1.

ML is an application of artificial intelligence that permits systems to automatically learn from data and make decisions without human assistance. ML enables systems to learn from experience and recover without being explicitly programmed. ML can mechanize and increase the efficiency of intelligent traffic congestion control systems (TCCSs) while reducing travelling costs more efficiently and accurately in a reliable way [19]. Data fusion is a process of combining data from one and multiple sources with insufficient raw data to gather precise, comprehensive, and unified entity information. At the decision level, fusion is used to produce a single decision after combining decisions from multiple sources to create a more intelligent decision on action. By taking the data patterns of various algorithms, the fusion of decision-making data and ML can help make better choices [20].

In [22], researchers suggested a real-time TMS composed of a small system of roadside units, junction units, and mobile units that determine the time of traffic lights to avoid gridlock development. This system also includes a web-based application for drivers of vehicles that uses data from real-time traffic monitoring to show the current traffic flow so that approaching vehicles take alternate routes to help reduce the gridlock. The limitations of this technique are a lack of performance and a secure mechanism for communication between nodes.

Reference [23] used IoT and sensing technology to create a framework for real-time traffic monitoring. Ultrasonic sensors

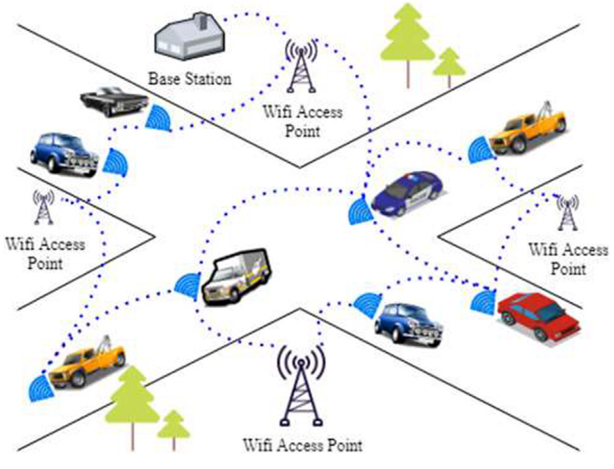

Fig. 1. Architecture of a VN [21].

were utilized to monitor traffic status in lanes. The controller collects this information from sensors and processes it. The processed data are then sent to the server via a Wi-Fi module. The traffic signal control system, which is based on perceiving traffic status in lanes, controls traffic. If a road has a significant amount of traffic, then it receives the highest priority, which means it takes a long time for vehicles to travel, and it will be given a long green-signal time. This proposed system is reliable, easy, and inexpensive. This presented system has limited calculation capacity and needs the proper execution of instructions by highly skilled programmers.

In [24], researchers developed a platform focused on Wireless Sensor Networks (WSNs) to acquire, fuse, and store city traffic data. The extended city intelligent transportation system is more versatile and efficient than other current city transportation systems. It is impossible to use WSNs for high-speed communication because they are designed for low-speed applications and are too expensive to build. In [25], the authors developed a vehicle traffic control mechanism via Usage-based Insurance (UBI) and smartphone-based measurement methods. This architecture is designed to model, perceive, and monitor traffic flow. This system has seven layers, starting with physical smartphones and servers and ending with the overall business strategy at the top.

In [26], researchers developed a model using a Random Forest Classification (RFC) algorithm of ML to construct a model for traffic congestion state perception. The RFC has high robustness, high efficiency, and a predictive accuracy of 87.5%. In addition, the generalization error is short and can be foreseen efficiently. The limitations of this research are that other machine learning techniques may provide more robust and accurate results. In [27], the authors developed a method for identifying road traffic congestion using GPS, a webcam, and an opinion poll. The sliding window technique extracted the vehicle's movement patterns and fed them into an Artificial Neural Network (ANN) and J48. The J48 model performs better with 91.29% accuracy, which is lower than that of the proposed FITCCS-VN using machine learning techniques.

In [28], the authors developed an Unmanned Aerial Vehicle (UAV)-based traffic monitoring system using a Convolutional Neural Network (CNN). The camera on the UAV captured traffic images, and the system had a 91.67% accuracy based on the traffic situation. In [29], the authors developed a data-fusion-based TCCS based on a CNN and Long Short-Term Memory (LSTM) frameworks. The CNN was used to classify spatial data. LSTM for historical data had a 92.3% and 7.7% accuracy and miss rate, respectively, whereas ML approaches may achieve better results. Different machine learning techniques such as computational intelligence approaches, fuzzy systems [30,31,32,33,34,35,36], swarm intelligence [37], evolutionary computing [38], soft computing [39], genetic algorithms [40], neural networks [41], and deep extreme learning machines [42] have proven to be credible solutions for smart cities [43,44,45], smart health [46,47,48], and wireless communication [49,50] by different researchers in different research domains.

3. Limitations of previous work

This paper presents a fusion-based intelligent traffic congestion control system for vehicular networks using machine learning techniques. Traffic congestion and the number of traffic accidents have increased rapidly. Proposed model solved the problem of short delays in communication between vehicles and roadside units, smooth traffic flow, and road safety by intelligent transportation systems. The aim of this study is to provide innovative services to drivers that enable a view of traffic flow and the volume of vehicles available on the road remotely, intending to avoid traffic jams. This study develops a fusion-based intelligent traffic congestion control system for vehicular networks using machine learning technology to alleviate traffic congestion. This contribution offers pathways to improve traffic flow and decrease congestion.

Table 1 shows that in the research mentioned above, there are several limitations including a lack of performance, the need for a secure communication mechanism between nodes [22], limited computational capacity [23], and lower accuracy [26,27,28,29].

The proposed techniques play a vital role in providing a secure communication mechanism between nodes by using a VN architecture, improving performance through a preprocessing layer to mitigate noisy data and achieving higher accuracy, higher execution capacity, and more robust decision-making by including fused ML techniques.

4. Proposed fusion-based intelligent traffic congestion control system

In this research, an intelligent TCCS is used for smart cities to monitor and control traffic congestion using machine learning techniques. This research proposed a FITCCS-VN using ML techniques. Figs. 2 and 3 present a complete picture of the proposed FITCCS-VN in which data are collected via IoV-enabled devices. This system allows signals from one junction to send and update data to another junction. Subsequently, the sensory layer receives data from sensors, and then these sensing values are passed through the preprocessing, training, performance, and validation phases.

Fig. 2 shows that VN has become a prominent network model with the development of communication technology. VN enables vehicles to broadcast with V2V and exchanges information with the Vehicle to Infrastructure (V2I). The collection of sensory data from wireless communication vehicles instead of costly cellular communication with the help of VN thus becomes possible. First, data can be assembled from roadside base stations via V2V or V2I communication. Then, the base station (server) can transfer the collected data directly to the data centre to further monitor and route traffic congestion.

Fig. 2 also depicts a proposed model in which the traffic congestion data sensed from the VN are received from the sensory layer and passed through preprocessing training and performance lay-

Table 1
Comparison of Previous Published Works with Proposed FITCCS-VN.

Authors	Preprocessing Layer	Use of VN	Decision-Making	Fused ML techniques
A. Saikar et al., 2017 [22]	No	No	No	No
V.S. Nagmode et al., 2017 [23]	No	No	Yes	No
Y. Liu et al., 2017 [26]	No	No	Yes	No
T. Thianniwet et al., 2010 [27]	No	No	Yes	No
L. Jian et al., 2010 [28]	Yes	No	Yes	No
S. Khan et al., 2021 [29]	Yes	No	Yes	Yes
Proposed FITCCS-VN	Yes	Yes	Yes	Yes

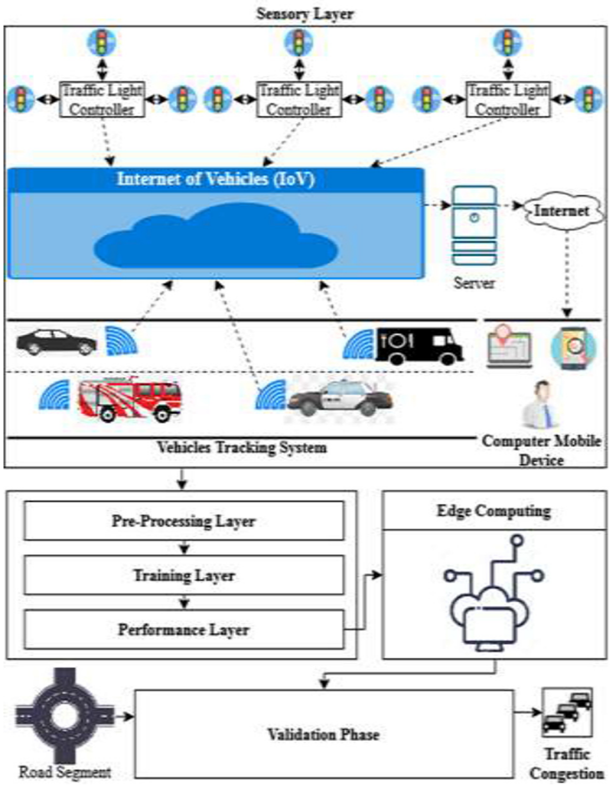

Fig. 2. Proposed FITCCS-VN using ML techniques.

ers. After the performance layer, the output is forwarded to edge computing. In the validation phase, the learned data are imported from edge computing to further predict whether there is traffic congestion.

As shown in Fig. 3, training and validation are the two phases of the proposed FITCCS-VN using ML techniques. Each step is further divided into stages. The 1st phase collects the dataset from various traffic control sensors installed on multiple VNs. To implement the proposed research model, a prelabelled VN dataset is selected. This dataset contains 2282 instances and 515 features, of which 514 attributes are independent and one, the output class, is dependent. The next layer is preprocessing, which mitigates the noisy data using moving averages and normalization. Then, the preprocessed data are divided into 70% training and 30% testing datasets. After this process, the training data are sent to the training layer, whereas the testing dataset is stored in edge computing. A classification process is performed in the training layer to predict the traffic congestion using both ML techniques [ANN and support Vector Machine (SVM)]. Each neuron has a function of activation, such as $\mathrm{f}(\mathrm{x}) = \mathrm{Sigmoid}(\mathrm{x})$ in the hidden layer. The sigmoid function for input and the hidden layer of the proposed FITCCS-VN can appear as.

$$\text{ӌ} = \frac{1}{1 + e^{-\left(p_2 + \sum_{j=1}^{n}\left(\upsilon_{j\kappa} * \frac{1}{1+e^{-p_1+\sum_{i=1}^{\omega}\left(\omega_{ij} * \acute{r}_i\right)}}\right)\right)\psi_\kappa}}, where\, j = 1, 2, 3 \ldots n \& \kappa = 1, 2, 3 \ldots e \quad (1)$$

$$\text{ȸ}_{\text{ӌ}} = \frac{1}{1 + \mathrm{e}^{-\left(p_2 + \sum_{j=1}^{n}\left(\upsilon_{j\text{ӌ}} * \frac{1}{1+e^{-\mathrm{p}_1+\sum_{\mathrm{i}=1}^{\text{É}-}\left(\omega_{\mathrm{ij}} * \acute{r}_{\mathrm{i}}\right)}}\right)\right)\psi_{\text{ӌ}}}} \text{ where } \mathrm{j} = 1, 2, 3 \cdots \mathrm{n} \,\&\, \text{ӌ} = 1, 2, 3 \cdots \acute{r}$$

In the above equation, $\acute{r}_i$, ω_{ij}, p_1, υ_{jk}, and p_2 represent the input features, weights amongst the i^{th} input and j^{th} hidden layer neurons, bias of hidden layers, weights between the jth hidden layer and ӌ^{th} output layer neurons, and the bias of output layer, respectively. These are listed in Table 2.

The minimum mean square error can be calculated as below:

$$A = \frac{1}{2}\sum_{\text{ӌ}}\left(\tau_{\text{ӌ}} - \text{ȸ}_{\text{ӌ}}\right)^2 \quad (2)$$

where $\tau_{\text{ӌ}}$ shows the estimated output, and $out_{\text{ӌ}}$ is a deliberate output. Both layers' weights in change can be calculated as below:

$$\Delta W \propto -\frac{\partial A}{\partial W}$$

$$\Delta \upsilon_{j,\text{ӌ}} = -\epsilon \frac{\partial A}{\partial v_{j,\text{ӌ}}} \quad (3)$$

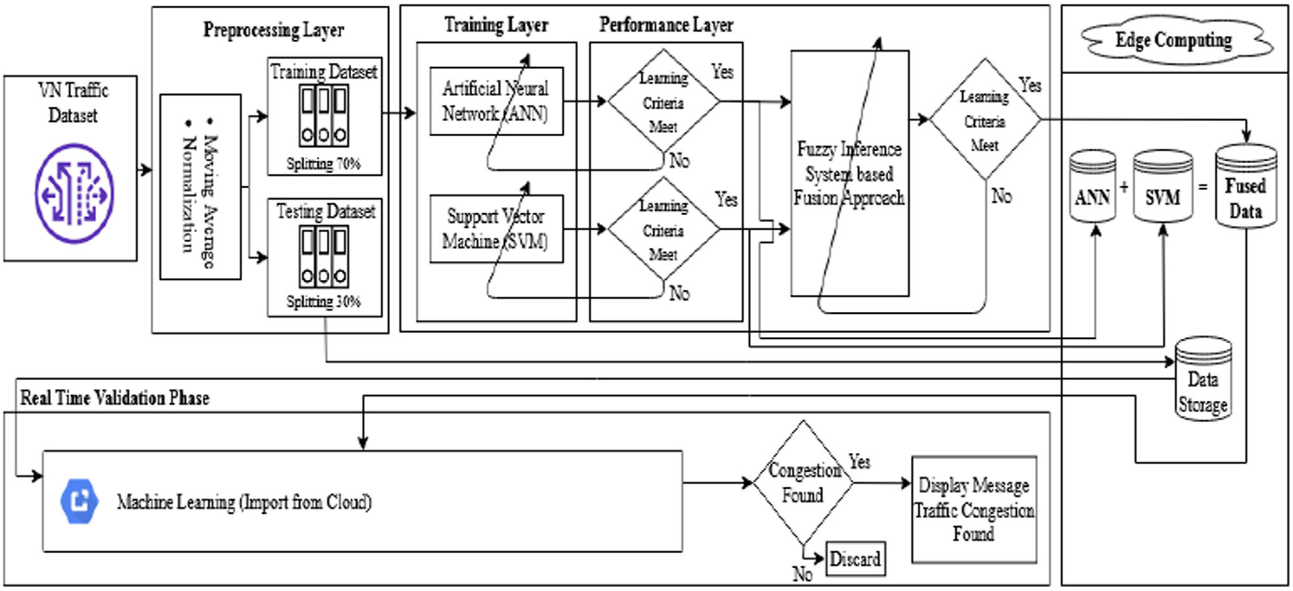

Fig. 3. Flowchart of Proposed FITCCS-VN using ML techniques.

Table 2
Dataset structure [52].

Sr. No.	Feature	Datatype
1	region_id	Integer
2	region_name	String
3	road_category_name	String
4	road_category_description	String
5	total_link_length_km	Float
6	total_link_length_miles	Float
7	pedal_cycles	Integer
8	two_wheeled_motor_vehicles	Integer
9	cars_and_taxis	Integer
10	buses_and_coaches	Integer
11	lgvs	Integer
12	all_hgvs	Integer
13	all_motor_vehicles	Integer

Eq. (3) can be written as.

$$\Delta \upsilon_{j,ʯ} = -\epsilon \frac{\partial A}{\partial ȸ_{ʯ}} \times \frac{\partial ȸ_{ʯ}}{\partial \psi_k} \times \frac{\partial \psi_k}{\partial v_{j,ʯ}} \tag{4}$$

The above equation after simplification can be written as.

$$\Delta \omega_{i,j} = \epsilon \left[\sum_{ʯ} (\tau_{ʯ} - ȸ_{ʯ}) \times ȸ_{ʯ}(1 - ȸ_{ʯ}) \times (\nu_{j,ʯ}) \right] \times ȸ_j(1 - ȸ_j) \times \acute{r}_i$$

$$\Delta \omega_{i,j} = \epsilon \left[\sum_{ʯ} (\tau_{ʯ} - ȸ_{ʯ}) \times ȸ_{ʯ}(1 - ȸ_{ʯ}) \times (\nu_{j,ʯ}) \right] \times ȸ_j(1 - ȸ_j) \times \acute{r}_i \Delta\omega_{i,j} = \epsilon \left[\sum_{ʯ} \xi_{ʯ}(v_{j,ʯ}) \right] \times ȸ_j(1 - ȸ_j) \times \acute{r}_i$$

$$\Delta \omega_{i,j} = \epsilon \xi_j \acute{r}_i \tag{5}$$

where

$$\xi_j = \left[\sum_{ʯ} \xi_{ʯ}(v_{j,ʯ}) \right] \times ȸ_j(1 - ȸ_j)$$

The hidden and output layers are presented in Eq. (6), which updates the weight and bias.

$$v^{+}_{j,ʯ} = v_{j,ʯ} + \lambda_F \Delta \upsilon_{j,ʯ} \tag{6}$$

Updating the weight and bias among the input layer and hidden layer is presented in Eq. (7):

$$\omega^{+}_{i,j} = \omega_{i,j} + \lambda_F \Delta \omega_{i,j} \tag{7}$$

λ_F denotes the learning rate of FITCCS-VN using ML techniques. The convergence of FITCCS-VN using ML techniques depends upon the careful selection of λ_F.

As we know, during SVM the line equation is.

$$ӿ = ʜʮ + ς \tag{8}$$

In Eq. (8), 'ʜ' represents the line slope and 'ς' the intersect. Hence,

$$ʜʮ - ӿ + ς = 0$$

Let $\bar{ʈ} = (ʮ, ӿ)^T$ and $\bar{ʧ} = (ʜ, -1)$. Then, the equation becomes.

$$\overrightarrow{ʧ.}\ \bar{ʈ} + ς = 0 \tag{9}$$

This equation comes from two-dimensional vectors. However, Eq. (9), defined as the hyperlane, performs for any number of dimensions. The direction of a vector $\bar{ʈ} = (ʮ, ӿ)^T$ is $ʧ$ and is distinct as.

$$ʧ = \frac{ʮ}{||ʈ||} + \frac{ӿ}{||ʈ||} \tag{10}$$

where

$$||ʈ|| = \sqrt{ʮ^2_+ ӿ^2_+ \cdots\cdots\cdots..ʈ^2_\zeta}$$

As we know that

$$\cos(\theta) = \frac{ʮ}{||ʈ||} \mathit{and} \cos(\mu) = \frac{ӿ}{||ʈ||}$$

Eq. (10) can also be written as.

$$ʧ = (\cos(\theta),\ \cos(\mu))$$

$$\overrightarrow{ʧ.}\ \overrightarrow{ʈ} = ||ʧ||\ ||ʈ|| \cos(\theta)$$

$$\theta = \overset{\hat{A}'}{\upsilon} - \mu$$

$$\cos(\theta) = \cos\left(\overset{\hat{A}'}{\upsilon} - \mu\right) = \cos(\overset{\hat{A}'}{\upsilon}) \cos(\mu) + \sin\left(\overset{\hat{A}'}{\upsilon}\right) \sin(\mu)$$

$$= \frac{\vartheta}{||ʧ||} \frac{ʮ}{||ʈ||} + \frac{\alpha}{||ʧ||} \frac{ӿ}{||ʈ||} = \frac{\vartheta ʮ + \alpha ӿ}{||ʧ|| ||ʈ||}$$

$$ʧ.ʈ = ||ʧ|| ||ʈ|| \left[\frac{\vartheta ʮ + \alpha ӿ}{||ʧ|| ||ʈ||} \right]$$

$$\overrightarrow{ʧ.}\ \overrightarrow{ʈ} = \sum_{i=1}^{\zeta} ʧ_i ʈ_i \tag{11}$$

The dot product can be compared using Eq. (11) for ζ dimensional vectors:

Let

$$B = M(ʧ.ʈ + ς)$$

If sign (B) > 0, then this is appropriately classified; and if sign (B) < 0, then it is imperfectly classified.

Calculate f on a training dataset by dataset Π:

$$B_i = M_i(ʧ.ʈ + ς)$$

þ is the functional margin of the dataset.

$$þ = \min_{i=1\cdots..Ƭ} B_i$$

When comparing hyperplanes, one through the largest þ will be chosen. þ is the geometric margin of the dataset. The goal is to discover an optimal hyperplane, which means finding the optimal hyperplane values of ʧ and B. Lagrangian function:

$$\breve{A}(ʧ, ς, \mu) = \frac{1}{2} ʧ.ʧ - \sum_{i=1}^{Ƭ} \mu_i [M : (ʧ.ʈ + ς) - 1]$$

$$\nabla_{ʧ} \breve{A}(ʧ, ς, \mu) = ʧ - \sum_{i=1}^{Ƭ} \mu_i M_i ʈ_i = 0 \tag{12}$$

$$\nabla_{ς} \breve{A}(ʧ, ς, \mu) = -\sum_{i=1}^{Ƭ} \mu_i M_i = 0 \tag{13}$$

From Eqs. (16) and (17), we get.

$$ʧ = \sum_{i=1}^{Ƭ} \mu_i M_i ʈ_i \text{ and } \sum_{i=1}^{Ƭ} \mu_i M_i = 0 \tag{14}$$

while substituting the Lagrangian function Ă:

$$\text{ʧ}(\mu,\varsigma)=\sum_{i=1}^{\text{Ꚍ}}\mu_i-\frac{1}{2}\sum_{i=1}^{\text{Ꚍ}}\sum_{j=1}^{\text{Ꚍ}}\mu_i\mu_j M_i M_j\,\text{ʈ}_i\text{ʈ}_j$$

Thus,

$$\max_{\mu}\sum_{i=1}^{\text{Ꚍ}}\mu_i-\frac{1}{2}\sum_{i=1}^{\text{Ꚍ}}\sum_{j=1}^{\text{Ꚍ}}\mu_i\mu_j M_i M_j\,\text{ʈ}_i\text{ʈ}_j \tag{15}$$

subject to $\mu_i \geq 0, i=1\cdots.\text{Ꚍ}, \sum_{i=1}^{\text{Ꚍ}}\mu_i M_i = 0$.

The Lagrangian multiplier technique is extended to Karush-Kuhn-Tucker (KKT) situations due to inequalities in the constraints. KKT's complimentary status states that

$$\mu_i\left[M_i(\text{ʧ}_i.\text{ʈ}^*+\varsigma)-1\right]=0 \tag{16}$$

ʈ^* denotes the optimal point.

μ is the positive value in addition to μ because the additional aspects are ≈ 0.

Thus,

$$M_i((\text{ʧ}_i.\text{ʈ}^*+\varsigma)-1)=0 \tag{17}$$

The points near the hyperplane are called support vectors. Equation (17) states that

$$\text{ʧ}-\sum_{i=1}^{\text{Ꚍ}}\mu_i M_i\,\text{ʈ}_i=0$$

$$\text{ʧ}=\sum_{i=1}^{\text{Ꚍ}}\mu_i M_i\,\text{ʈ}_i \tag{18}$$

To compute the value of ς we get.

$$M_i((\text{ʧ}_i.\text{ʈ}^*+\varsigma)-1)=0 \tag{19}$$

In Equation (19), multiply both sides by M to get.

$$M_i^2((\text{ʧ}_i.\text{ʈ}^*+\varsigma)-M_i)=0$$

where $M_i^2 = 1$

$$((\text{ʧ}_i.\text{ʈ}^*+\varsigma)-M_i)=0$$

$$\varsigma=M_i-\text{ʧ}_i.\text{ʈ}^* \tag{20}$$

Then

$$\varsigma=\frac{1}{\text{я}}\sum_{i=1}^{\text{я}}(M_i-\text{ʧ}.\text{ʈ}) \tag{21}$$

я is the number of support vectors. On one occasion, the hyperplane will create perceptions. The hypothesis function is.

$$c(\text{ʧ}_i)=\begin{bmatrix}1\mathit{if}\,\text{ʧ}.\text{ʈ}+\varsigma>0\\0\mathit{if}\,\text{ʧ}.\text{ʈ}+\varsigma\leq 0\end{bmatrix} \tag{22}$$

The hyperplane is classified as traffic congestion (positive), and the point below the hyperplane is classified as no traffic congestion (negative). Therefore, the primary purpose of the SVM algorithm is to perceive a hyperplane that can disperse the data precisely, in addition to the best need to be found, which is often called a hyperplane.

After the training layer, the output is sent to the performance layer to predict the traffic congestion based on the accuracy and miss rate regardless of whether the learning criteria are met. If the answer is 'NO', then the training layer is updated, and so on. However, if the answer is 'YES', then the output of both approaches is stored in ANN and SVM-based edge databases separately and sent to the Fuzzy Inference System (FIS)-based fusion approach. The FIS-based fusion approach further checks and is updated if the learning criteria are not met. In the case of yes, the output is stored in the fused database on edge computing.

In the validation phase, the test data stored in edge computing and the learned patterns are imported from the edge database and referred to the ML techniques to predict whether traffic congestion is found. If the answer is 'No', then the process is discarded; and if the answer is 'Yes', then the message indicates that traffic congestion has been found.

The fusion approach using ML techniques develops and applies fuzzy logic to optimized classification algorithms. ML techniques (ANN and SVM) produce logic using fuzzy rules. The conditional statements to create fuzzy logic are given below.

1. If ANN is No and SVM is No, then FITCCS-VN is No
2. If ANN is No and SVM is Yes, then FITCCS-VN is Yes
3. If ANN is Yes and SVM is No, then FITCCS-VN is Yes
4. If ANN is Yes and SVM is Yes, then FITCCS-VN is Yes

Fig. 4 illustrates that if the value of SVM lies between 60 and 100 and ANN lies between 60 and 100, then FITCCS-VN is good (yellow). If SVM lies between 40 and 60 and ANN lies between 40 and 60, then FITCCS-VN is satisfactory (green). If SVM lies between 0 and 40 and ANN lies between 0 and 40, then FITCCS-VN is satisfactory (blue).

Fig. 5 demonstrates that if the value of ANN is no and SVM is no, then the proposed FITCCS-VN is no. Fig. 6 shows that if the value of ANN is yes and SVM is no, then the proposed FITCCS-VN is yes. Fig. 7 demonstrates that if the value of ANN is yes and SVM is yes, then the proposed FITCCS-VN is yes.

5. Simulation results

In this proposed FITCCS-VN using ML techniques, a TCCS is implemented on a dataset [52]. The data were divided randomly into 70% training (1879 samples) and 30% validation (403 samples). The proposed system calculated the output using multiple statistical measures, as described in Equations (23) through (31).

$$Sensitivity=\frac{\sum TruePositive}{\sum ConditionPositive} \tag{23}$$

$$Specificity=\frac{\sum TrueNegative}{\sum ConditionNegative} \tag{24}$$

$$Accuracy=\frac{\sum TruePositive+\sum TrueNegative}{\sum TotalPopulation} \tag{25}$$

$$Miss-Rate=1-Accuracy \tag{26}$$

$$Fallout=\frac{\sum FalsePositive}{\sum ConditionNegative} \tag{27}$$

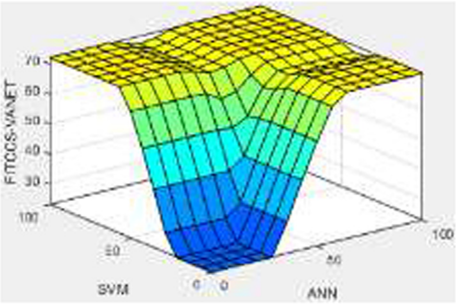

Fig. 4. Rule surface of Proposed FITCCS-VN based on SVM and ANN.

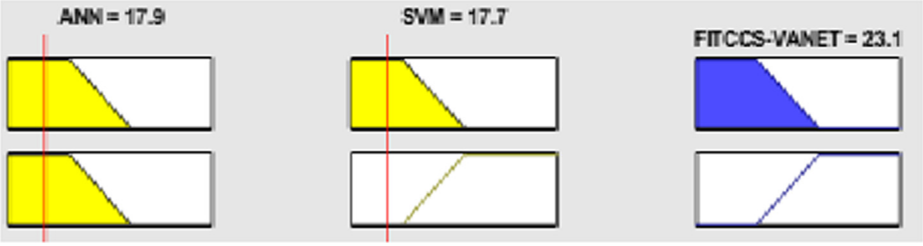

Fig. 5. Lookup diagram of Proposed FITCCS-VN (No).

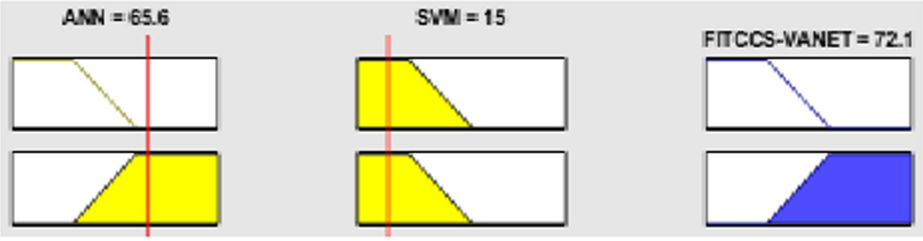

Fig. 6. Lookup diagram of the Proposed FITCCS-VN (Yes).

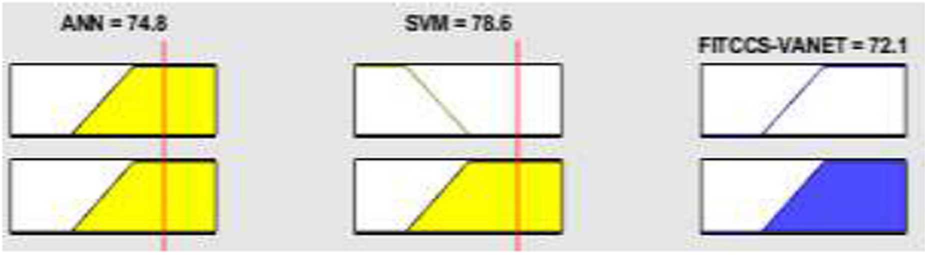

Fig. 7. Lookup diagram of Proposed FITCCS-VN (Yes).

$$\mathit{LikelihoodPositiveRatio} = \frac{\sum \mathit{TruePositiveRatio}}{\sum \mathit{FalsePositiveRatio}} \tag{28}$$

$$\mathit{LikelihoodNegativeRatio} = \frac{\sum \mathit{TruePositiveRatio}}{\sum \mathit{FalsePositiveRatio}} \tag{29}$$

$$\mathit{PositivePredictiveValue} = \frac{\sum \mathit{TruePositive}}{\sum \mathit{PredictedConditionPositive}} \tag{30}$$

$$\mathit{NegativePredictiveValue} = \frac{\sum \mathit{TrueNegative}}{\sum \mathit{PredictedConditionNegative}} \tag{31}$$

Tables 3, 4, 5, and 6 show the training and validation of ML techniques (ANN and SVM) in terms of accuracy and miss rate. In addition to comparisons, the various statistical measures used for performance are deliberate from diverse metrics named as accuracy, sensitivity, specificity, miss-rate, fall-out, Likelihood Positive Ratio (LR+), Likelihood Negative Ration (LR-), Precision, and negative predictive value. By contrast, the True Positive Rate (TPR) is expressed as sensitivity, True Negative Rate (TNR) as specificity, False Negative Rate (FNR) as miss-rate, False-Positive Rate (FPR) as fall-out, and Positive Predictive Value (PPV) as precision.

Table 3
Training of proposed FITCCS-VN using the ML technique (ANN) when identifying traffic congestion.

Proposed Model Training			
Input	Total samples (1879)	Output	
	Expected output	Predicted Positive	Predicted Negative
	293 Positive	True Positive (TP) 237	False Negative (FN) 56
	1586 Negative	False-Positive (FP) 31	True Negative (TN) 1555

Table 4
Validation of proposed FITCCS-VN using ML technique (ANN) when identifying traffic congestion.

Proposed Model Validation			
Input	Total samples (403)	Output	
	Expected output	Predicted Positive	Predicted Negative
	53 Positive	True Positive (TP) 40	False Negative (FN) 13
	350 Negative	False-Positive (FP) 11	True Negative (TN) 339

Table 5
Training of proposed FITCCS-VN using ML technique (SVM) when identifying traffic congestion.

Proposed Model Training			
Input	Total samples (1879)	Output	
	Expected output	Predicted Positive	Predicted Negative
	293 Positive	True Positive (TP) 228	False Negative (FN) 65
	1586 Negative	False-Positive (FP) 56	True Negative (TN) 1530

Table 6
Validation of proposed FITCCS-VN using ML technique (SVM) when identifying traffic congestion.

Proposed Model Validation			
Input	Total samples (403)	Output	
	Expected output	Predicted Positive	Predicted Negative
	53 Positive	True Positive (TP) 35	False Negative (FN) 18
	350 Negative	False-Positive (FP) 21	True Negative (TN) 329

Table 3 shows the proposed FITCCS-VN using ML techniques within the training phase. A total of 1879 samples are used within the training separated into 293 and 1586 positive and negative samples, respectively. It is ascertained that 237 samples are appropriately predicted as positive and that there is no traffic congestion, but 56 records are inaccurately predicted as negative, which means there is traffic congestion. Similarly, a total of 1586 samples are collected, where negative indicates traffic congestion, in which 1555 samples are appropriately predicted as negative, meaning there is traffic congestion, and 31 samples are wrongly predicted as positive, which means there is no traffic congestion.

Table 4 shows the proposed FITCCS-VN using ML techniques within the validation phase. A total of 403 samples were used within the validation, divided into 53,350 positive and 53,350 negative samples. It is ascertained that 40 samples are appropriately predicted as positive and that there is no traffic congestion, but 13 records are inaccurately predicted as negative, which indicates traffic congestion. Correspondingly, a total of 350 samples are collected, where negative indicates traffic congestion, in which 339 samples are appropriately predicted as negative, meaning there is traffic congestion, and 11 samples are wrongly predicted as positive, which means there is no traffic congestion.

Table 5 shows the proposed FITCCS-VN using ML techniques within the training phase. A total of 1879 samples were used within the training, separated into 293 and 1586 positive and negative samples, respectively. It is ascertained that 228 samples are correctly predicted as positive and that there is no traffic congestion, but 65 records are incorrectly predicted as negative, which indicates traffic congestion. Similarly, a total of 1586 samples are collected, where negative indicates traffic congestion, in which 1530 samples are appropriately predicted as negative, meaning there is traffic congestion, and 56 samples are wrongly predicted as positive, which means there is no traffic congestion.

Table 6 shows the proposed FITCCS-VN using ML techniques within the validation phase. A total of 403 samples were used within the validation, divided into 53,350 positive and 53,350 negative samples. It is ascertained that 35 samples are appropriately predicted as positive and that there is no traffic congestion, but 18 records are inaccurately predicted as negative, which means there is traffic congestion. Correspondingly, a total of 350 samples are collected, where negative means there is traffic congestion, in which 329 samples are appropriately predicted as negative, which means there is traffic congestion, and 21 samples are wrongly perceived as positive, which means there is no traffic congestion.

Table 7 shows the performance of the proposed FITCCS-VN using ML techniques in terms of accuracy, sensitivity, specificity, miss rate, and precision within the training and validation phases w.r.t. ANN and SVM, respectively. This indicates that the proposed system using the ANN approach within training provides 95.4%, 80.8%, 98.0, 4.6%, and 88.4% in terms of accuracy, sensitivity, specificity, miss rate, and precision. Within validation, the proposed system provides 94.0%, 75.4%, 96.8%, 6.0%, and 78.4% accuracy, sensitivity, specificity, miss rate, and precision, respectively. In addition, more statistical measures of the proposed system can predict values such as fall-out, likelihood positive ratio, likelihood negative ratio, and negative predictive value within training. The results were 0.019, 42.526, 0.047, and 96.5% and validations of 0.031, 24.323, 0.062, and 96.3%, respectively.

This indicates that the proposed system with the SVM approach within training provides 93.5%, 77.8%, 96.4%, 6.5%, and 80.2% in terms of accuracy, sensitivity, specificity, miss rate, and precision. Within validation, the proposed system provides 90.3%, 66.0%, 94.0%, 9.7%, and 62.5% accuracy, sensitivity, specificity, miss rate, and precision, respectively. In addition, more statistical measures of the proposed system can predict values such as fall-out, likelihood positive ratio, likelihood negative ratio, and negative predictive value within training. These results are 0.035, 22.045, 0.281, and 95.9% and validation of 0.06, 11.005, 0.361, and 94.8%, respectively.

Moreover, the simulation results of the fused ML techniques of the proposed FITCCS-VN are listed in Table 8. Twenty random values were deployed using the fusion of ANN and SVM, of which 19 were precise according to human decision-making principles of FITCCS-VN. In contrast, one matter was thought to be low but was indicated as normal per the proposed FITCCS-VN using ML techniques and incorrectness. The probability of correctness or accuracy of the proposed FITCCS-VN using ML techniques is found to be 95%, and the miss rate is 5%.

Table 9 elaborates on the simulation of the proposed FITCCS-VN with fused machine learning techniques. The accuracies and miss rates of the ANN, SVM, and fusion-based FITCCS-VN are 94%/6%, 90.3%/9.7%, and 95%,/5%, respectively.

Table 10 shows a performance comparison of the proposed FITCCS-VN using the ML techniques with previous approaches

Table 7
Performance evaluation of proposed FITCCS-VN using ML techniques in training and validation using diverse statistical measures.

		Accuracy (%)	Sensitivity TPR (%)	Specificity TNR (%)	Miss-Rate FNR (%)	Fall-out FPR	LR + ve	LR -ve	PPV (Precision) (%)	NPV (%)
ANN	Training	95.4	80.8	98.0	4.6	0.019	42.526	0.047	88.4	96.5
	Validation	94.0	75.4	96.8	6.0	0.031	24.323	0.062	78.4	96.3
SVM	Training	93.5	77.8	96.4	6.5	0.035	22.045	0.281	80.2	95.9
	Validation	90.3	66.0	94.0	9.7	0.06	11.005	0.361	62.5	94.8

Table 8
Fusion results of proposed FITCCS-VN using ML techniques (ANN and SVM).

	ANN	SVM	The proposed FITCCS-VN using ML techniques	The human expert decision of FITCCS-VN	Probability of correctness	Probability of errors
1	12.4 (N)	20.5 (N)	23.1 (N)	No	1	0
2	28.9 (N)	7.73 (N)	23.1 (N)	No	1	0
3	17.9 (N)	65 (Y)	72.1 (Y)	Yes	1	0
4	10.6 (N)	13.2 (N)	23.1 (N)	No	1	0
5	24.3 (N)	22.3 (N)	23.1 (N)	No	1	0
6	10.6 (N)	64.1 (Y)	72.1 (Y)	Yes	1	0
7	74.8 (Y)	13.2 (N)	72.1 (Y)	Yes	1	0
8	81.2 (Y)	6.82 (N)	72.1 (Y)	Yes	1	0
9	72 (Y)	68.6 (Y)	72.1 (Y)	Yes	1	0
10	12.4 (N)	20.5 (N)	23.1 (N)	No	1	0
11	24.3 (N)	7.73 (N)	23.1 (N)	No	1	0
12	77.5 (Y)	15 (N)	72.1 (Y)	Yes	1	0
13	17.9 (N)	79.5 (Y)	72.1 (Y)	Yes	1	0
14	91.3 (Y)	90.5 (Y)	72.1 (Y)	Yes	1	0
15	24.3 (N)	66.8 (Y)	72.1 (Y)	Yes	1	0
16	83.9 (Y)	65 (Y)	72.1 (Y)	Yes	1	0
17	77.5 (Y)	13.2 (N)	72.1 (Y)	Yes	1	0
18	77.5 (Y)	66.8 (Y)	72.1 (Y)	Yes	1	0
19	93.1 (Y)	79.5 (Y)	72.1 (Y)	Yes	1	0
20	27.1 (N)	84.1 (Y)	23.1 (N)	Yes	0	1

Table 9
ANN and SVM results of proposed FITCCS-VN.

	ANN	SVM	Fusion based FITCCS-VN
Accuracy (%)	94.0	90.3	95
Miss Rate (%)	6.0	9.7	5

Table 10
Comparison results of proposed FITCC-VN using ML techniques with literature.

		Accuracy (%)	Miss-Rate (%)
Thianniwet T. et al. (2010), [27]	Training	91.29	8.71
	Validation	90.11	9.89
Jian, L. et al., (2019) [28]	Training	91.67	8.33
	Validation	90.5	9.5
Khan, Sulaiman et al., (2021) [29]	Training	93.6	6.4
	Validation	92.3	7.7
Pushpi and Dilip Kumar (2018) [51]	Training	91.2	8.8
	Validation	90.6	9.4
Proposed system using ML technique (ANN)	Training	95.4	4.6
	Validation	94.0	6.0
Proposed system using ML technique (SVM)	Training	93.5	6.5
	Validation	90.3	9.7
Proposed FITCCS-VN using ML techniques	Validation	95	5

[27,28,29,51]. It is indicated that the proposed techniques provide accurate results compared to previously published approaches.

6. Conclusion

A Vehicular Network (VN) is a self-organized, service-oriented, multipurpose communication network that enables communication between vehicles and roadside infrastructure for message exchange. In a dense traffic scenario, the load generated by the traffic may exceed the road's capacity, causing traffic congestion. This research proposed a fusion-based ITCCS-VN using ML techniques to assemble data from an IoV-enabled VN, and then evaluated it intelligently to predict and control traffic congestion. Evaluation of the simulation results indicated that the proposed FITCCS-VN using ML techniques exhibits 95% accuracy and a 5% miss rate, which are better than those of previous approaches [27,28,29,51]. In the future, the proposed system accuracy may be improved by using federated learning and Alexnet.

References

[1] Albino V, Berardi U, Dangelico RM. Smart cities: Definitions, dimensions, performance, and initiatives. J Urban Technol 2015;22(1):3–21.

[2] Ravish R, Swamy SR. Intelligent Traffic Management: A Review of Challenges, Solutions, and Future Perspectives. Transp Telecommun 2021;22(2):163–82.

[3] Marshall WE, Dumbaugh E. Revisiting the relationship between traffic congestion and the economy: a longitudinal examination of US metropolitan areas. Transportation 2020;47(1):275–314.

[4] Banishree G. Intelligent Mobility for Minimizing the Impact of Traffic Incidents on Transportation Networks. Augmented Intelligence Toward Smart Vehicular Applications, 2020;27(3):175–194.

[5] Li G, Lai W, Sui X, Li X, Qu X, et al. Influence of traffic congestion on driver behavior in post-congestion driving. Accid Anal Prev 2020;141(1):105508.

[6] Eze EC, Zhang SJ, Liu EJ, Eze JC. Advances in vehicular ad-hoc networks (VANETs): Challenges and road-map for future development. Int J Autom Comput 2016;13(1):1–18.

[7] Ramesh K, Lakshna A, Renjith PN. Smart Traffic Congestion model in IoT-A Review. In: 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA). p. 651–8.

[8] Jayapal C, Roy SS. Road traffic congestion management using VANET. In: International Conference on Advances in Human Machine Interaction (HMI). p. 1–7.

[9] El Zorkany M, Yasser A, Galal AI. Vehicle to vehicle "V2V" communication: Scope, importance, challenges, research directions and future. Open Transp J 2020;14(1):86–98.

[10] Wan S, Xu X, Wang T, Gu Z. An intelligent video analysis method for abnormal event detection in intelligent transportation systems. IEEE Trans Intell Transp Syst 2020;22(7):4487–95.

[11] Nwankwo W, Olayinka AS, Ukhurebor KE. The Urban Traffic Congestion Problem in Benin City and the Search for an ICT-improved Solution. Int J Sci Technol 2019;8(1):12–9.

[12] Sadhukhan P, Gazi F. An IoT based intelligent traffic congestion control system for road crossings. In: In 2018 International Conference on Communication, Computing and Internet of Things (IC3IoT). p. 371–5.

[13] Ullo SL, Sinha GR. Advances in smart environment monitoring systems using IoT and sensors. Sensors 2020;20(11):1–18.

[14] Qureshi KN, Din S, Jeon G, Piccialli F. Internet of vehicles: Key technologies, network model, solutions and challenges with future aspects. IEEE Trans Intell Transp Syst 2020;22(3):1777–86.

[15] Günay FB, Öztürk E, Çavdar T, Hanay YS. Vehicular ad hoc network (VANET) localization techniques: a survey. Arch Comput Methods Eng 2021;28 (4):3001–33.

[16] Yan G, Rawat DB. Vehicle-to-vehicle connectivity analysis for vehicular ad-hoc networks. Ad Hoc Netw 2017;58(1):25–35.

[17] Vijayakumar V, Joseph KS. Adaptive load balancing schema for efficient data dissemination in Vehicular Ad-Hoc Network VANET. Alexandria Eng J 2019;58 (4):1157–66.

[18] Yang H, Jia Z, Xie G. Delay-bounded and cost-limited RSU deployment in urban vehicular ad hoc networks. Sensors 2018;18(9):2764–85.

[19] Meena G, Sharma D, Mahrishi M. Traffic Prediction for Intelligent Transportation System using Machine Learning. In: 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things (ICETCE). p. 145–8.

[20] Meng T, Jing X, Yan Z, Pedrycz W. A survey on machine learning for data fusion. Inform Fusion 2020;57(1):115–29.

[21] Badis H, Rachedi A. Modeling tools to evaluate the performance of wireless multi-hop networks. Model Simul Comput Netw Syst 2015;18(2):653–82.

[22] Saikar A, Parulekar M, Badve A, Thakkar S, Deshmukh A. TrafficIntel: Smart traffic management for smart cities. In: International Conference on Emerging Trends & Innovation in ICT (ICEI). p. 46–50.

[23] Nagmode VS, Rajbhoj SM. An intelligent framework for vehicle traffic monitoring system using IoT. Int Conf Intell Comput Control (I2C2), pp. 1-4, 2017.

[24] Wang H. Wireless sensor networks for an extended city intelligent transportation system. Int J Adv Comput Technol 2011;3(5):300–7.

[25] Händel P, Ohlsson J, Ohlsson M, Skog I, Nygren E. Smartphone-based measurement systems for road vehicle traffic monitoring and usage-based insurance. IEEE Syst J 2013;8(4):1238–48.

[26] Liu Y, Wu H. Prediction of road traffic congestion based on random forest. In: 10th International Symposium on Computational Intelligence and Design (ISCID), vol. 2, no. 1, pp. 361-364, 2017.

[27] Thianniwet T, Phosaard S, Pattara-Atikom W. Classification of road traffic congestion levels from vehicle's moving patterns: a comparison between artificial neural network and decision tree algorithm. Electro Eng Comput Technol 2010;8(23):261–71.

[28] Jian L, Li Z, Yang X, Wu W, Ahmad A, Jeon G. Combining unmanned aerial vehicles with artificial-intelligence technology for traffic-congestion recognition: electronic eyes in the skies to spot clogged roads. IEEE Consum Electron Mag 2019;8(3):81–6.

[29] Khan S, Nazir S, García-Magariño I, Hussain A. Deep learning-based urban big data fusion in smart cities: Towards traffic monitoring and flow-preserving fusion. Comput Electr Eng 2021;89(1):1–11.

[30] Aftab S, Alanazi S, Ahmad M, Khan MA, Fatima A, et al. "Cloud-Based Diabetes Decision Support System Using Machine Learning Fusion. Comput Mater Continua 2021;68(1):1341–57.

[31] Khan QTA, Abbas S, Khan MA, Fatima A, Alanazi S, et al. Modelling intelligent driving behaviour using machine learning. Comput Mater Continua 2021;68 (3):3061–77.

[32] Saleem M, Khan MA, Abbas S, Asif M, Hassan M, et al. Intelligent FSO link for communication in natural disasters empowered with fuzzy inference system. In: 2019 International Conference on Electrical, Communication, and Computer Engineering (ICECCE). p. 1–6.

[33] Tabassum N, Ditta A, Alyas T, Abbas S, Alquhayz H, et al. Prediction of cloud ranking in a hyperconverged cloud ecosystem using machine learning. Comput Mater Continua 2021;67(1):3129–41.

[34] Alghamdi MA, Khan MFN, Khan AK, Khan I, Ahmed A, et al. PV model parameter estimation using modified fpa with dynamic switch probability and step size function. IEEE Access 2021;9(4):42027–44.

[35] Nadeem L, Azam MA, Amin Y, Ghamdi MA, Chai KK, et al. Integration of D2D, network slicing, and MEC in 5G cellular networks: Survey and challenges. IEEE Access 2021;9(7):37590–612.

[36] Khan AH, Khan MA, Abbas S, Siddiqui SY, Saeed MA, et al. Simulation, modeling, and optimization of intelligent kidney disease predication empowered with computational intelligence approaches. Comput Mater Continua 2021;67(2):1399–412.

[37] Ahmad G, Alanazi S, Alruwaili M, Ahmad F, Khan MA, et al. Intelligent ammunition detection and classification system using convolutional neural network. Comput Mater Continua 2021;67(2):2585–600.

[38] Shoaib B, Javed Y, Khan MA, Ahmad F, Majeed M, et al. Prediction of time series empowered with a novel srekrls algorithm. Comput Mater Continua 2021;67 (2):1413–27.

[39] Khan F, Khan MA, Abbas S, Athar A, Siddiqui SY, Khan AH, et al. Cloud-based breast cancer prediction empowered with soft computing approaches. J Healthcare Eng 2020;2020:1–16.

[40] Nadeem MW, Goh HG, Khan MA, Hussain M, Mushtaq MF, et al. Fusion-based machine learning architecture for heart disease prediction. Comput Mater Continua 2021;67(2):2481–96.

[41] Batool T, Abbas S, Alhwaiti Y, Saleem M, Ahmad M, Asif M, et al. Intelligent Model Of Ecosystem For Smart Cities Using Artificial Neural Networks. Intell Automat Soft Comput 2021;29(3):513–25.

[42] Ahmed A, Khan MFN, Khan I, Alquhayz H, Khan MA, et al. A novel framework to determine the impact of time varying load models on wind dg planning. IEEE Access 2021;9(2):11342–57.

[43] Khan AI, Kazmi SAR, Atta A, Mushtaq MF, Idrees M, et al. Intelligent cloud-based load balancing system empowered with fuzzy logic. Comput Mater Continua 2021;67(1):519–528.

[44] Siddiqui SY, Naseer I, Khan MA, Mushtaq MF, Naqvi RA, et al. Intelligent breast cancer prediction empowered with fusion and deep learning. Comput Mater Continua 2021;67(1):1033–49.

[45] Alhaidari F, Almotiri SH, Al Ghamdi MA, Khan MA, Rehman A, et al. Intelligent software-defined network for cognitive routing optimization using deep extreme learning machine approach. Comput Mater Continua 2021;67 (1):1269–85.

[46] Khan MA, Umair M, Saleem MA, Ali MN, Abbas S. CDE using improved opposite based swarm optimization for MIMO systems. J Intell Fuzzy Syst 2019;37 (1):687–92.

[47] Khan MA, Umair M, Saleem MA. GA based adaptive receiver for MC-CDMA system. Turkish J Electr Eng Comput Sci 2015;23(Sup. 1):2267–77.

[48] Nadeem MW, Ghamdi MAA, Hussain M, Khan MA, Khan KM, Almotiri SH, et al. Brain tumor analysis empowered with deep learning: A review, taxonomy, and future challenges. Brain Sci 2020;10(2):118.

[49] Atta A, Abbas S, Khan MA, Ahmed G, Farooq U. An adaptive approach: Smart traffic congestion control system. J King Saud Univ-Comput Inform Sci 2020;32(9):1012–9.

[50] Khan MA, Abbas S, Rehman A, Saeed Y, Zeb A, Uddin MI, et al. A machine learning approach for blockchain-based smart home networks security. IEEE Netw 2021;35(3):223–9.

[51] Pushpi R, Shaw KD. Artificial neural networks approach induced by fuzzy logic for traffic. J Eng Technol 2018;6(2):127–41.

[52] www.kaggle.com/arashnic/road-trafic-dataset?select=region_traffic.csv.