Journal of Scientific & Industrial Research
Vol. 84, May 2025, pp. 544-555
DOI: 10.56042/jsir.v84i5.8489

Application of an Integrated Fog-IoT Framework to a Smart Traffic Surveillance Management System

Jui-Pin Yang
College of Marine Resources and Engineering, National Penghu University of Science and Technology, Taiwan

Received 02 February 2024; revised 14 November 2024; accepted 19 March 2025

In addressing urban traffic management, a prevalent approach involves the installation of surveillance cameras along roadways. However, this approach raises two critical concerns. The primary issue lies in the potential for negligence caused by extended periods of manual monitoring. Hence, the prevailing trend in traffic management has shifted towards the adoption of intelligent traffic surveillance management systems. The second challenge pertains to the need for swift responses to traffic conditions, necessitating real-time and efficient management strategies. In this research paper, we present an integrated Fog-IoT framework for a Smart Traffic Surveillance Management System (STSMS). This framework leverages IoT devices and fog nodes for task processing, significantly enhancing overall system performance. The STSMS utilizes surveillance cameras to collect extensive traffic data. Subsequently, the collected data is rigorously analyzed to accurately assess congestion levels in adjacent areas. The system autonomously generates commands to control traffic signals, effectively coordinating neighboring signals, thereby achieving efficient and intelligent traffic management. Furthermore, the STSMS promptly communicates its findings to administrators, enabling proactive responses. For instance, it can dispatch notifications to nearby police stations, facilitating the allocation of personnel to alleviate traffic congestion. Clearly, the STSMS plays a pivotal role in the development of smart cities, not only by facilitating intelligent traffic management but also by optimizing resource utilization, including reductions in latency and network usage. To assess the effectiveness of the STSMS comprehensively, simulations were conducted using the iFogSim tool. The experimental results demonstrate unequivocally that the Fog-IoT-based STSMS significantly reduces latency and network usage compared to cloud-based frameworks. These findings underscore the transformative potential of the STSMS in revolutionizing urban traffic management, thereby advancing the vision of efficient and intelligent smart cities.

Keywords: Fog computing, iFogSim tool, Internet of Things, Smart city, Urban traffic management

Introduction

The Internet of Things (IoT) represents a vast ecosystem of heterogeneous devices, playing a pivotal role in collecting diverse data for a multitude of advanced applications. IoT's pervasive influence extends across various domains, encompassing smart factories, intelligent cities, environmental surveillance, and healthcare monitoring systems.[1,2] As IoT applications gain widespread traction, a surge of research endeavors has emerged, aimed at enhancing their feasibility and applicability. This surge is in direct response to the growing number of IoT devices, which brings to the critical challenge of effectively detecting malicious activities and ensuring robust security.

In this context, a novel hybrid classification approach has been introduced, tailored to detect multi-class attacks in IoT networks.[3] This multifaceted approach involves a sequence of stages: firstly, principal component analysis is applied to extract pertinent features; next, linear discriminant analysis is utilized to pinpoint critical features; and finally, a combination of neural network and support vector machine classifiers is harnessed to bolster detection rates while curbing false alarms. The outcome is a solution that not only trims computational complexity but also fortifies intrusion detection capabilities.

IoT devices conventionally rely on sensors to monitor environmental conditions and convey collected data for advanced applications. However, as the number of IoT devices swells, this can result in network congestion, resulting in elevated latency, heightened packet loss, and diminished throughput. To tackle this predicament head-on, an innovative deep neural network-Restricted Boltzmann machine (DNN-RBM) algorithm has been put forth, specifically designed for predicting network congestion. Impressively, this approach showcases a remarkable 95% accuracy rate in congestion

*Author for Correspondence
E-mail: juipinyang@gmail.com

prediction, achieved by scrutinizing congested nodes within sensor-laden IoT environments.[4] The DNN-RBM ingeniously optimizes weight parameters by synergizing with the Restricted Boltzmann Machine (RBM) algorithm, emerging as a promising solution for congestion prediction.

Typically, cloud computing serves as the primary infrastructure for processing tasks generated by IoT devices. Nevertheless, dealing with the colossal volume of data generated in the cloud can lead to pronounced communication overheads. To address this concern, numerous studies have advocated for fog-based architectures, with the primary aim of augmenting system performance metrics such as efficiency, security, and cost-effectiveness.[5,6] These forward-looking fog-based architectures incorporate a range of features, including virtualization, edge computing, and IoT services, all deployed at the outermost edge of the telecommunications network. Furthermore, some propose Software-Defined Networking (SDN)-based fog computing architectures, explicitly designed to support mobility.[6] These solutions employ route optimization algorithms that seamlessly cater to user mobility requirements. The advent of fog computing holds substantial promise in reducing network traffic, minimizing the strain on network bandwidth, and, most notably, offering significantly reduced latency compared to conventional cloud computing.[7,8] Fog computing is undeniably well-suited for IoT environments, given its capacity to streamline network operations, improve scalability, and mitigate latency-related challenges.[9]

The network bandwidth plays a dominant role for real-time applications. In another word, the fog computing is effective to minimize latency.[10] The distance is usually remote between cloud servers and the IoT devices, which results in high transmission latency and consumes a lot of network bandwidth. Apparently, cloud servers are unsuitable to handle latency-sensitive applications.[11] To overcome such disadvantages, fog computing was considered as a complementary of the cloud computing, which works as an intermediate component between the cloud servers and the IoT devices.[12,13] The fog computing makes computing, storage and network resources spatially closer to the IoT devices and hence it is useful to reduce latency and bandwidth consumption. The fog computing consists of a few fog networks which offload intensive tasks from the IoT devices.[14] Moreover, a fog network is composed of several fog nodes with limited resources.[15] Accordingly, they are capable of handling partial tasks redirecting from the IoT devices located in their radius of coverage. Efficient offloading within the fog computing paradigm is the key solution that contributes to maximizing resource utilizations and satisfies the QoS requirements of real-time tasks.

Simulation tools play a pivotal role in evaluating various performance metrics under diverse network environments, ranging from software-defined networking to named data networking and fog-based systems. Traditionally, tools like NS-2, TOSSIM, and OMNeT++ have served as industry standards for network protocol and algorithm evaluation. However, these tools were not explicitly designed to address the unique challenges posed by fog environments. The iFogSim is a cutting-edge simulation tool tailored explicitly for fog-based environments, specializing in the evaluation of latency, energy consumption, and network usage across diverse fog network infrastructures and applications.[16] This open-source simulation tool meticulously models the elements comprising fog environments: fog nodes, IoT devices, cloud resources, sensors, network links, and applications. It also takes power monitoring and resource management into account, making it a comprehensive solution for fog-based simulations. Furthermore, researchers have employed iFogSim in collaborative approaches to compare and analyze network usage and latency against baseline methods.[17] One noteworthy approach leveraged iFogSim to design fog devices, sensor classes, physical topologies, and actuator classes, thus enhancing the tool's utility in modeling and assessing fog-based systems.[18]

Given the prominent role of simulation tools in assessing system performance, the choice of a simulation platform becomes critical. The ability of iFogSim to model the intricacies of IoT and fog environments makes it an ideal choice for evaluating the performance metrics of the Smart Traffic Surveillance Management System (STSMS). The inadequacies of the STSMS within a cloud-based framework are evident, leading to heightened latency and extensive network usage. To effectively address these limitations, this paper introduces the Fog-IoT framework for the STSMS.

This paper contributes significantly in three key dimensions:

1. A groundbreaking Fog-IoT framework for the STSMS is proposed, wherein fog nodes and IoT

devices are meticulously co-designed to enhance overall system performance.

2. The STSMS demonstrates its ability to efficiently alleviate traffic congestion in real time by reducing latency and network usage through the synergy of IoT devices and fog nodes.
3. A rigorous series of simulations is conducted to comprehensively evaluate the performance metrics of the Fog-IoT framework in comparison to a cloud-based framework for the STSMS. Experimental results conclusively demonstrate that the Fog-IoT framework significantly reduces latency and network usage compared to the cloud-based framework.

Literature Review

Compared to cloud-based systems, fog networks inherently possess constrained resources, encompassing limited network bandwidth, computing capabilities, and storage capacities.[19,20] In response to these limitations, caching mechanisms have been leveraged to optimize latency and bandwidth utilization. Additionally, proactive system remediation strategies have been implemented based on data load prediction.[21] The collaborative synergy between the cloud, which stores historical data, and fog networks, responsible for live data, has been explored to enhance predictive analytics and forecasting of essential metrics.[22]

In recent years, machine learning has emerged as a powerful tool to address these challenges. For instance, machine learning-based proposals have excelled in predicting user behaviors, aiding in proactive caching decisions that reduce latency and elevate the quality of experience.[23] Another critical challenge revolves around energy efficiency in resource management.[24] While offloading data processing to the cloud can reduce energy consumption, it simultaneously increases bandwidth usage and causes additional power consumption.[25] Therefore, adopting an integrated approach that considers energy efficiency in both cloud and fog networks becomes crucial when optimizing resource allocations, addressing energy consumption, performance, and Quality of Service (QoS) requirements.[26–28]

The versatility of resource utilization in fog networks, providing a range of data processing options, stands as a significant advantage. From a cloud standpoint, the effective caching and processing of offloaded data play a crucial role in mitigating network congestion and enhancing latency. Conversely, IoT devices gain advantages from data offloading schemes that transfer computational tasks to fog networks, thereby improving overall performance, particularly in terms of energy efficiency.[29,30] Moreover, the performance requirements for IoT devices, especially in real-time applications like online gaming, video conferencing, and financial trading, necessitate latency levels below tens of milliseconds. Addressing dynamic and rapidly changing environments requires efficient mobility management methods for mobile edge computing. Meeting the stringent demands of mobile edge computing necessitates an extensive deployment of small cells. However, the limited user ranges and frequent handovers lead to significant communication overheads.[31] Fog networks and IoT devices utilize a diverse range of communication technologies, from low-cost protocols to energy-efficient protocols, each with a unique impact on performance factors such as data processing capacity, service time, and latency.

The integration of fog computing and edge computing into distributed sites hosting multiple applications necessitates automated system construction methods.[32] Given the inherent scale and complexity of fog and edge systems, developing realistic prototypes becomes impractical. Furthermore, commercial service providers typically do not grant third-party control over infrastructure. Consequently, an effective testbed should consider complexity, resource diversity, and execution time.[33] Therefore, simulation frameworks emerge as indispensable tools for evaluating the performance of fog and edge systems, determining the feasibility and viability of proposed policies.[34]

IoT technologies serve as the foundation for constructing smart cities based on sensor infrastructures.[35] These smart cities integrate information and communication technologies to enhance management efficiency, delivering intelligent and optimized services.[36] IoT devices, being cost-effective components, significantly contribute to smart cities by enhancing intelligent traffic management, surveillance, air quality monitoring, and more. However, deploying IoT devices poses several performance challenges, including latency, bandwidth constraints, and limited resources. Furthermore, considerations regarding connectivity, data privacy, and security are paramount.

Challenges in technology integration for smart cities grow proportionally with the proliferation of IoT devices. Consequently, cloud-centric platforms prove impractical, lacking economic and sustainable models that fail to meet demands such as low latency and high availability.[37] To address these challenges, studies have proposed innovative approaches, such as the implementation of ARIMA models with optimized hyperparameters using grid search techniques for traffic flow predictions.[38] These approaches have yielded promising results by incorporating diverse datasets, including daily traffic flow patterns and morning and evening peak traffic flows. Additionally, hybridized evolutionary algorithms have been employed to fine-tune prediction models for mortality predictions in road traffic accidents in India, demonstrating their potential for enhancing safety and resource allocation.[39] The advent of 6G wireless technologies demands innovative routing protocols for MANETs and VANETs, especially in smart cities. This paper introduces the Multi-Metric Scoring Dynamic Source Routing (MMS-DSR), an enhancement of DSR tailored for 6G environments.[40] MMS-DSR integrates a CNN-LSTM-based beamforming algorithm to dynamically optimize beam directions, improving link reliability and throughput. It employs a multi-metric scoring mechanism to evaluate routes based on QoS parameters like latency, bandwidth, and reliability, leveraging Massive MIMO and IEEE 802.11ax for adaptive selection. Machine learning predictions enable proactive routing adjustments, making MMS-DSR a robust solution for high-mobility, dynamic networks in smart cities.

The literature reveals several gaps that highlight the need for the present study. Firstly, existing research on cloud-centric architectures fails to address the scalability and latency challenges posed by the growing number of IoT devices. Secondly, while fog networks are seen as a potential solution, resource management strategies in these networks are often limited, especially in optimizing QoS. Additionally, simulation frameworks for fog-based systems remain inadequate for modeling real-time in dynamic environments. Current smart city platforms relying on cloud computing also struggle with low latency and high availability. To fill these gaps, this study proposes a Fog-IoT Framework for Smart Traffic Surveillance Management Systems, which optimizes resource allocation, reduces latency, and enhances real-time performance, especially in congestion management.

Smart Traffic Surveillance Management System

The Smart Traffic Surveillance Management System (STSMS) employs sophisticated three-tier architecture, as visually depicted in Fig. 1. This architecture encompasses various essential components, including cameras, Traffic Signal Controllers (TSCs), Smart Traffic Surveillance (STS) devices, fog nodes, routers, and the cloud. Each layer plays a distinct role in the seamless operation of the STSMS.

The foundational layer, constituting the first tier, revolves around STS devices. Each STS device is a comprehensive unit comprising a camera and a TSC. These STS devices are strategically positioned on traffic signposts, providing a vantage point for capturing live traffic video feeds. These cameras serve as the eyes of the system, diligently recording traffic conditions. The captured video streams serve as the primary data source for assessing traffic congestion. The STS component takes charge of regulating traffic signals in response to detected congestion. Remarkably, the STS can be realized using versatile platforms such as Raspberry Pi, Arduino, ESP32, among others, offering flexibility and scalability.

Ascending to the second tier, we encounter the pivotal fog nodes. These fog nodes serve as orchestration hubs, overseeing the management and provision of advanced resources to the STS devices. Each fog node efficiently manages a cluster of STS devices, ensuring optimized performance and coordination within their designated areas of influence. The third and final tier involves routers,

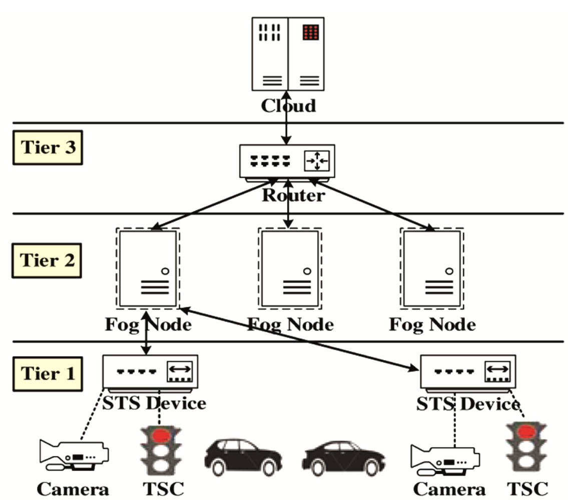

Fig. 1 — Three-tier architecture of the smart traffic surveillance management system

serving as the communication backbone connecting the fog nodes with the cloud. These routers facilitate the transmission of data between fog nodes and the cloud, ensuring that valuable insights and information can be swiftly relayed to the cloud for further processing and analysis.

The operational flow within this architecture unfolds as follows: Initially, the cameras fulfill their role by capturing video footage of road traffic. Subsequently, these video feeds undergo sophisticated analysis by the STS devices and fog nodes to assess the level of traffic congestion accurately. Importantly, the cameras are seamlessly integrated into the STS devices, ensuring a streamlined data flow. The STS device acts as a vital bridge, facilitating seamless communication between the fog nodes, cameras, and TSCs. This connectivity plays a pivotal role in orchestrating timely responses to traffic conditions. The fog nodes, acting as intermediaries, communicate with the cloud via the routers. The data, collected and analyzed by the fog nodes, is then transmitted to the cloud for further processing and storage. Each fog node operates within a specified geographical area, overseeing a cluster of STS devices. In the event of traffic congestion within its jurisdiction, the fog node acts intelligently by sending precise instructions to the relevant STS devices to adjust traffic signals accordingly. This responsive action significantly contributes to traffic management and alleviation of congestion in real-time.

While the cloud boasts substantial resources in terms of storage, computing power, and bandwidth, it is inherently challenged by latency constraints due to the considerable physical distance it covers. Moreover, relying solely on the cloud for data processing would impose a significant burden on network bandwidth. However, the introduction of the intermediate fog network layer proves to be a strategic solution for mitigating latency challenges. Within the presented framework, the primary workload and processing tasks are efficiently handled by the STS devices and fog nodes, ensuring that the system operates optimally and meets the low latency requirements of smart traffic management.

This well-structured three-tier architecture underscores the synergy of devices and technologies, emphasizing the critical role of each component in achieving efficient, real-time traffic surveillance and management within the STSMS framework. System interactions from the perspective of the STS devices are thoughtfully illustrated in Fig. 2. The intricate process begins with the initialization of the STS device, as it establishes direct connections with both the camera and TSC. Subsequently, it forges a communication link with the corresponding fog node, initiating a seamless flow of data and commands within the system.

The camera diligently captures video footage, promptly relaying it to the STS device. Here, the magic unfolds as the STS device undertakes object detection and traffic identification tasks with precision and efficiency. Object detection assumes the crucial responsibility of estimating the number of vehicles present, while traffic identification delves into various parameters, including vehicle count, car speed, congestion duration, and more. These factors collectively form the basis for assessing the levels of traffic congestion. Crucially, the compilation of statistics and STS parameters is deftly managed by the astute fog node. Upon receiving this comprehensive data, the fog node takes on the role of a traffic maestro, orchestrating the best strategy to tackle the congestion menace. With a strategic plan in hand, the fog node disseminates control messages to the relevant STS devices, setting in motion a symphony of actions to regulate traffic signals through their TSCs.

Furthermore, the related fog nodes dutifully update the results to the cloud, ensuring that this valuable data is harnessed for future analysis and decision-making. The cloud, serving as the nerve center of the operation, doesn't rest on its laurels. Upon receiving the results, it promptly notifies the administrator, who steps into action. Armed with critical insights, the administrator can view the results and, when necessary, mobilize manpower to intervene and manage congested traffic effectively.

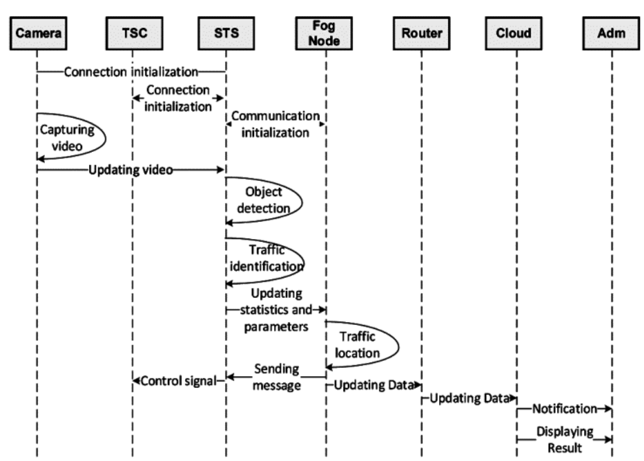

Fig. 2 — System interaction from the perspective of the STS devices

An illustrative representation of the cloud-based framework, meticulously integrated into the STSMS using iFogSim, is shown in Fig. 3. This framework seamlessly connects STS devices denoted as STS device 1-1 to STS device 1-n to the cloud via router 1. The system is designed with a total of m routers, each intricately linking with n STS devices. Importantly, each STS device constitutes a harmonious pairing of a camera and a TSC, exemplified by Camera 1-1 and TSC 1-1's affiliation with STS device 1-1.The operational flow unfolds as follows: Initially, the cameras diligently capture traffic video footage, channeling it through the STS devices and routers to reach the cloud. However, a direct connection between the cameras and the cloud, while feasible, raises concerns of heightened latency and substantial network bandwidth consumption.

Within the cloud, a sophisticated processing pipeline takes command, analyzing the incoming video streams with precision and speed. The analysis results are subsequently channeled to the traffic control center, where administrators harness this valuable information to orchestrate advanced traffic control strategies. In parallel, the cloud generates automatic traffic control signals, efficiently dispatching them to the relevant STS devices. This seamless communication ensures real-time traffic management, optimizing traffic flows and bolstering efficiency.

By intricately weaving together cloud-based processing, real-time analytics, and advanced control strategies, this framework elevates the STSMS to new heights of effectiveness and responsiveness. This holistic approach, driven by automated analysis and strategic decision-making, serves as a cornerstone for modern traffic management systems, promising safer, more efficient cities in the era of urban mobility transformation.

A comprehensive visual representation of the Fog-IoT-based framework, meticulously integrated into the STSMS using the iFogSim simulation tool, is shown in Fig. 4. Within this framework, there exist m routers and p fog nodes, each playing pivotal roles in the system's architecture. Importantly, a single fog node is tasked with overseeing the operations of n STS devices, ensuring efficient management and coordination. Much like its cloud-based counterpart, each STS device within this framework comprises a camera and a TSC, forming an inseparable unit. Both

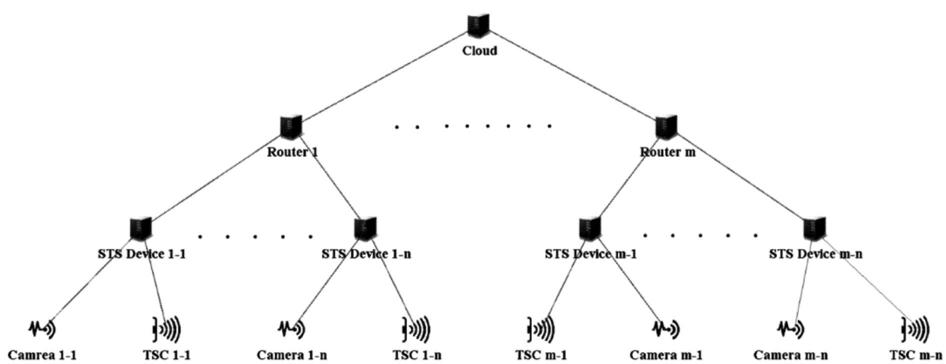

Fig. 3 — Cloud-based framework

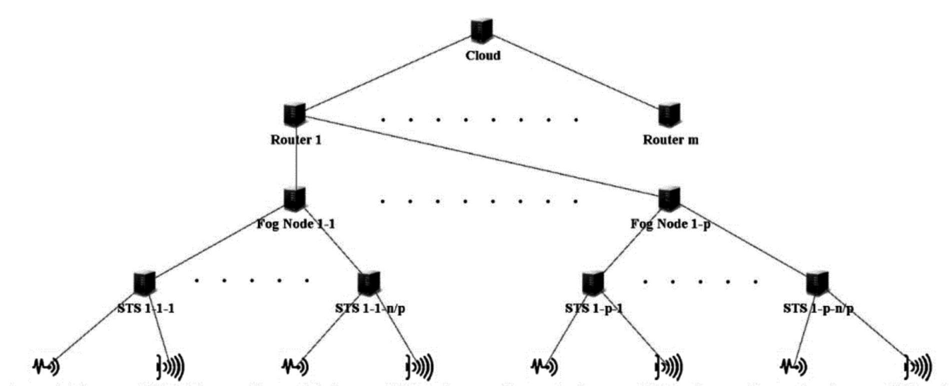

Fig. 4 — Fog-IoT based framework

the cloud-based and Fog-IoT-based frameworks have been thoughtfully designed and implemented to facilitate a rigorous evaluation of their respective latency and network usage. These assessments have been conducted through comprehensive simulations utilizing the powerful capabilities of the iFogSim simulation tool. Through these simulations, valuable insights have been gained, shedding light on the performance, efficiency, and potential of each framework. These insights will undoubtedly inform the ongoing development and optimization of the STSMS, contributing to its evolution as a cutting-edge solution for intelligent traffic management.

A comprehensive application model for the STSMS, wherein vertices symbolize distinct modules and edges signify the interdependencies of data among these modules, has been shown in Fig. 5. This model encapsulates the intricate architecture of the STSMS, comprising six fundamental modules: Camera, Object Detector, Traffic Identifier, Traffic Locator, Display Result, and TSC Control. Each module's specifications are elucidated as follows:

(1) Camera Module: This vital component is entrusted with capturing real-time traffic videos, subsequently transmitting them to the Object Detector module.
(2) Object Detector Module (OD): The Object Detector module plays a pivotal role in the system, conducting real-time detection of vehicular objects and forwarding pertinent statistics to the Traffic Identifier module. This data triggers the identification of traffic conditions.
(3) Traffic Identifier Module (TI): The Traffic Identifier module assumes the critical task of conducting real-time analysis and identification of vehicular objects. It aggregates data from various Object Detector modules to ascertain the levels of traffic congestion impacting nearby roadways.
(4) Traffic Locator Module (TL): Drawing upon information gleaned from Traffic Identifier modules, the Traffic Locator module undertakes a comprehensive analysis of affected roadways. Subsequently, the module devises traffic signal control plans to alleviate congestion. These prediction reports and control plans are transmitted to the Display Result module for further management and analysis. Furthermore, the Traffic Locator module autonomously dispatches control signals to the TSC Control modules.
(5) Display Result Module: This module serves as the repository for storing results within the cloud. In the event of traffic congestion, automatic notifications are triggered, alerting administrators. Upon receiving these notifications, administrators access the results, considering potential measures for congestion mitigation, such as dispatching law enforcement personnel for on-site traffic management.
(6) TSC Control Module: When traffic control signals are received, the TSC Control module springs into action, sending control messages to the relevant Traffic Signal Controllers (TSCs). This coordinated effort serves to alleviate traffic congestion swiftly and efficiently.

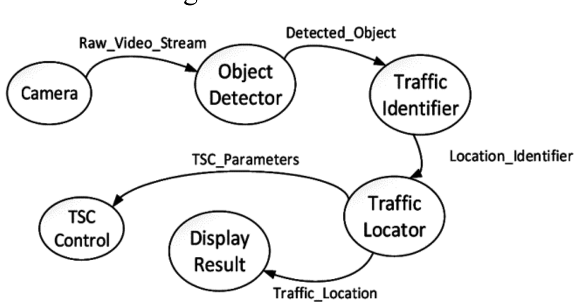

Fig. 5 — Application model for the STSMS

These modules operate collaboratively to ensure the timely monitoring, prediction, and management of traffic within smart cities. In the cloud-based framework, the OD module is deployed within the STS device, while the remaining modules reside in the cloud infrastructure. In the Fog-IoT framework, two distinct deployment scenarios are considered: Fog and Fog-IoT. The Fog scenario involves deploying the OD module within the STS device, with the remaining modules situated in the fog node. Conversely, the Fog-IoT scenario entails the deployment of both the OD and TI modules within the STS device, while the TL module resides in the fog node. This strategic consideration of module deployment underscores the comprehensive nature of the STSMS design and its adaptability to varying deployment scenarios.

Latency stands as a paramount consideration, particularly in applications entailing stringent real-time requirements. The inclusion of fog nodes within the system architecture serves as a strategic approach to mitigate latency concerns. These fog nodes significantly reduce the frequency of interactions with the cloud by performing offload computing at the network's edge, thereby facilitating rapid responses to IoT devices. This notable role of fog nodes in latency reduction cannot be overstated. Essentially, the fog nodes play a pivotal role in minimizing latency. They assume responsibility for crucial tasks, including object detection and traffic identification, in close proximity to IoT devices. Each fog node is exclusively allocated to a specific geographical area, ensuring an abundance of resources dedicated to expediting application processing. This localized approach empowers the STSMS to effectively and promptly manage traffic congestion.

The absence of fog nodes within the system would necessitate the transmission of substantial volumes of data to the cloud, inevitably resulting in escalated network usage and exacerbated latency. The strategic deployment of fog nodes circumvents these challenges, fostering an environment conducive to efficient, real-time traffic management. Within the iFogSim framework, each module is conceptually represented as a virtual machine, specializing in the processing of a specific type of tuple (i.e., task) received from the preceding module within the dataflow. Tuple forwarding between two modules can occur periodically or be initiated upon the reception of a tuple of a specific type. This intelligent orchestration streamlines the flow of data within the system, ensuring optimal communication and task management between modules. A comprehensive outline of the parameter settings governing tuple interactions across different modules can be found in Table 1. This strategic integration of fog nodes not only enhances the system's real-time capabilities but also bolsters its efficiency, ultimately contributing to more effective traffic management within the STSMS.

Results and Discussion

In this research endeavor, we harnessed the capabilities of iFogSim, a proficient simulation toolkit meticulously chosen for the assessment of the STSMS. Employing iFogSim affords us the means to comprehensively scrutinize diverse performance metrics, encompassing factors such as latency and network utilization. Within the simulation framework, the STSMS is meticulously replicated, comprising an assemblage of essential components including cloud infrastructure, routers, fog nodes, STS devices, cameras, and TSCs. To systematically investigate the influence of varying scenarios, we deliberately augmented the number of STS devices. A comprehensive exposition of the parameter configurations pertinent to the cloud, routers, fog nodes, and STS devices, as employed in the simulations, is presented in Table 2. The delineated parameters encompass a spectrum of crucial aspects, encompassing processing capability measured in Million Instructions Per Second (MIPS), random access memory (RAM) allocation, uplink and downlink bandwidth capacities (in megabytes), processing rate measured in million instructions processed per second, power consumption in both busy and idle states (in watts), among others. These settings are integral to the meticulous evaluation of the STSMS's performance under varying conditions.

Latency, a crucial concern in high-performance, real-time-demanding settings, demands diligent attention and mitigation. Fog computing, as a solution, offers a host of benefits in tackling this challenge. By steering away from frequent cloud interactions and instead orchestrating computations at the network's periphery, fog computing ensures expeditious responses to client devices, ultimately resulting in

Table 1— Parameter settings of Tuples

Tuple source	Tuple destination	Tuple cpu length (MIPS)	Tuple Nw length (Byte)
CAMERA	Object detector	1000	20000
Object detector	Traffic identifier	2000	2000
Traffic identifier	Traffic locator	1000	100
Traffic locator	Display result	500	2000
Traffic locator	Traffic control	100	100

Table 2 — Parameter settings for the STSMS

Parameter	Cloud	Router	Fog node	STS device
CPU length (MIPS)	4480000	50000	20000	1000
RAM (MB)	400000	20000	10000	1000
Uplink bandwidth (MB)	100	10000	10000	1000
Downlink bandwidth (MB)	10000	10000	10000	1000
Level	0	1	2	3
RatePerMIPS	0.01	0.0	0.0	0.0
Busy power (Watt)	1648	107.339	107.339	87.53
Idle power (Watt)	1332	83.4333	83.4333	82.44

reduced latency. Within this framework, the transmission of parking slot images finds its way to specialized fog nodes strategically positioned at the network's edge. Each fog node is exclusively assigned to a specific area, affording an ample reservoir of computational power for processing location-specific images and efficiently updating LED displays with real-time slot availability updates.

The quantification of $Latency$is achieved through the utilization of Eq. 1.[16]

$$Latency = \alpha + \mu + \omega \quad \ldots (1)$$

where, α represents the Tuple CPU Execution Delay involved in capturing images, μ accounts for the time needed to upload videos to the fog node for processing and storage, and ω encompasses the time taken to display processed data on the LED after undergoing processing at the fog node.

Optimizing these variables holds the potential to yield substantial reductions in latency, thereby significantly elevating real-time performance within fog computing environments. When traffic increases on the cloud server, only cloud resources are utilized. The escalation of traffic on the cloud server leads to heightened network utilization. Consequently, data transfer rates on the network decrease due to the increased traffic load. In the case of geographically distributed servers, a dedicated fog node is assigned to each geographical area to manage requests specific to that region, thereby improving the overall efficiency of resource allocation. This results in a decrease in network usage in those areas and an increased transmission rate for the remaining traffic, ultimately reducing network congestion. Network usage is quantified using Eq. 2.[16]

$$Network\ usage = Latency \times \text{tupleNWSize} \quad \ldots (2)$$

where, the tuple NW Size typically refers to the size of a data tuple transmitted over a network.

These results affirm our ability to meet the challenges posed by varying traffic loads and geographical distribution, ultimately enhancing the overall performance and responsiveness of the system, which is critical for ensuring a seamless and efficient experience for users. A comprehensive analysis of latency across diverse module deployment scenarios, encompassing Cloud, Fog, and Fog-IoT configurations, is presented in Fig. 6. The results gleaned from this analysis reveal noteworthy insights into the latency dynamics associated with each scenario. In the Cloud scenario, latency exhibits remarkable stability, hovering consistently at around 1000 milliseconds, seemingly impervious to fluctuations induced by an expanding array of STS devices.

Conversely, both the Fog and Fog-IoT scenarios chart a markedly different trajectory, characterized by a pronounced reduction in latency compared to the Cloud scenario. This favorable latency reduction can be attributed to the decentralized processing of tasks by IoT devices and the proficient fog nodes. As the number of STS devices scales up from 20 to 200, the Fog-IoT scenario emerges as the frontrunner, boasting even lower latency than the Fog scenario. This is primarily attributed to the more equitable distribution of processing responsibilities, alleviating the burden on fog nodes.

Nevertheless, an interesting observation emerges as the number of STS devices escalates to 300, resulting in a rapid upswing in latency within both the Fog and Fog-IoT scenarios. The underlying cause is the inadequacy of resources within the STS devices and fog nodes to accommodate the escalating workload. Consequently, certain modules necessitate transfer to and processing by the router, precipitating a substantial escalation in latency, albeit still remaining below the levels observed in the Cloud scenario. To address this issue, the resources of STS devices and fog nodes should be scaled up proportionally with the increase in the number of STS devices.

A comprehensive depiction of network usage comparisons across the same deployment scenarios explored in Fig. 6 is offered in Fig. 7. A notable observation gleaned from Fig. 7 underscores the prodigious bandwidth consumption inherent in the cloud scenario, primarily attributed to its comprehensive oversight of all data processing activities.

As we systematically augment the number of STS devices, a conspicuous surge in network usage

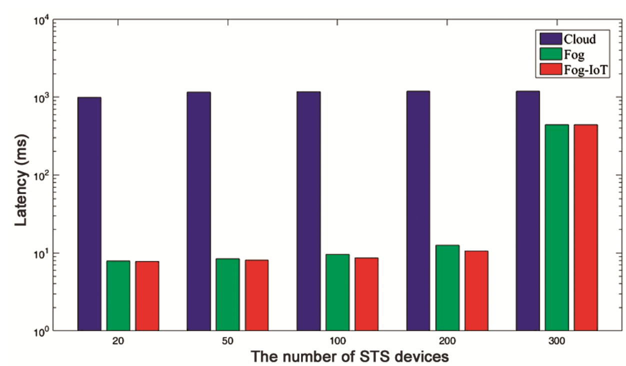

Fig. 6 — Comparisons of latency for different scenarios of module deployment

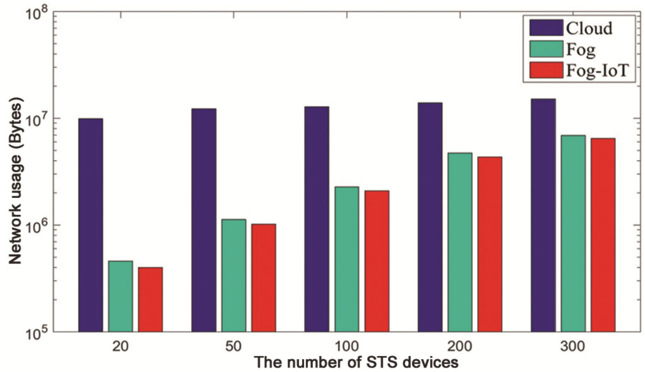

Fig. 7 — Comparisons of network usage for different scenarios of module deployment

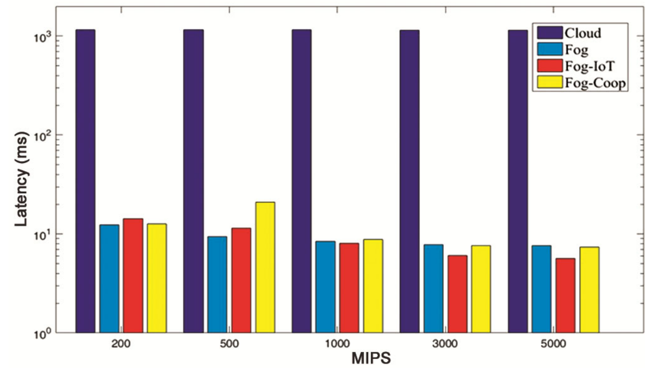

Fig. 8 — Comparisons of latency versus the STS device with different MIPS

becomes evident within both the Fog and Fog-IoT scenarios. This is unsurprising, as the increased volume of data generated necessitates greater bandwidth allocation to accommodate the augmented workload. However, a noteworthy trend emerges from the comparison between the Fog and Fog-IoT scenarios. The Fog-IoT scenario exhibits a distinctly more economical consumption of bandwidth when contrasted with the Fog scenario. This reduction in bandwidth consumption can be attributed to the optimized data transmission protocols and resource allocation strategies deployed within the Fog-IoT framework. In summation, the empirical findings corroborate the superior performance of the Fog-IoT scenario in terms of both latency and bandwidth usage, establishing it as a compelling alternative to the Fog scenario and a substantially more efficient choice when juxtaposed with the resource-intensive cloud scenario.

A comprehensive analysis of latency in relation to MIPS (Million Instructions Per Second) across various deployment scenarios, encompassing Cloud, Fog, Fog-IoT, and Fog-Coop configurations, is provided in Fig. 8. The Fog-Coop scenario represents a unique configuration wherein the OD module resides in the STS device, while the TL module is positioned within the fog node. Notably, both the STS device and fog node feature the TI module, with each TI module having a 50% probability of processing traffic identification tasks.

When the MIPS of the STS device is configured at 200, an interesting observation emerges. The latency of the Fog-IoT scenario surpasses that of both the Fog and Fog-Coop scenarios. This counterintuitive outcome can be attributed to the insufficient MIPS, prompting the TI module to autonomously migrate to the fog node. Consequently, this migration slightly inflates latency. However, as the MIPS of the STS device is elevated to 500, an intriguing shift in latency dynamics transpires. The Fog-Coop scenario exhibits heightened latency compared to both the Fog and Fog-IoT scenarios. This discrepancy arises because the TI module, situated within the STS device, does not relocate to the fog node, thereby elongating the processing duration.

Remarkably, once the MIPS of the STS device exceed the 500, the Fog-IoT scenario emerges as the frontrunner, boasting the lowest latency among all scenarios. In contrast, both the Fog and Fog-Coop scenarios manifest nearly identical latency profiles, indicative of the limited efficiency of distributed TI modules. In summation, these empirical findings underscore the instrumental role of the Fog-IoT framework in ameliorating latency and optimizing bandwidth utilization within the STSMS, affirming its status as a potent solution for enhancing system performance.

The use of iFogSim simulation software comes with several limitations. Firstly, iFogSim relies on abstract models to represent the system, which may not fully capture the complexities of real-world environments, such as hardware heterogeneity, dynamic network conditions, and non-ideal operational scenarios. This simplification can lead to discrepancies between simulated and actual system behavior. Secondly, while iFogSim provides estimates of resource consumption (e.g., computational and network resources), these estimates may not always be accurate, especially under high load or dynamic conditions, where real device performance may significantly differ from the simulation. Thirdly, iFogSim does not account for hardware failures or service interruptions, which are critical factors in real systems and can significantly affect performance. Finally, the simulation typically uses generated data traffic, which may not represent real-world traffic patterns, limiting its ability to accurately model the behavior of actual data flows in a real deployment. These limitations highlight the need for caution when generalizing simulation results to practical, real-world systems.

Conclusions

This study introduces an integrated Fog-IoT framework for the Smart Traffic Surveillance Management System (STSMS) in smart cities. The proposed framework effectively detects traffic congestion and enables timely, intelligent traffic management. Empirical results demonstrate that the Fog-IoT framework reduces latency and network usage significantly compared to conventional cloud-based systems. Additionally, the performance of the framework has been validated across various deployment scenarios. Future work will aim to enhance the framework by incorporating cooperative interactions among fog nodes, which are expected to further reduce latency and improve resource utilization. Load balancing across multiple fog nodes will also be prioritized to optimize overall system performance.

References

1 Dalipi F & Yayilgan S Y, Security and privacy considerations for IoT application on smart grids: Survey and research challenges, *IEEE Future IoT Cloud Workshops* (IEEE) 2016, 63–68.

2 Zhang D, Chen Z, Awad M K, Zhang N, Zhou H & Shen X S, Utility-optimal resource management and allocation algorithm for energy harvesting cognitive radio sensor networks, *IEEE J Select Areas Commun*, **34(12)** (2016) 3552–3565.

3 Choudhary S & Nishtha K, A hybrid classification approach for intrusion detection in IoT network, *J Sci Ind Res*, **80(9)** (2021) 809–816.

4 Rajesh K Y, Aastha M & Prem N, Data congestion prediction in sensors based IoT network, *J Sci Ind Res*, **80(12)** (2021) 1091-1095.

5 Ricard V, Victor L, Alessio G, Shuping P, Vittorio O & Luis V, TelcoFog: A unified flexible fog and cloud computing architecture for 5G networks, *IEEE Commun Mag*, **55(8)** (2017) 36–43.

6 Yuanguo B, Guangjie H, Chuan L, Qingxu D, Lei G & Fuliang L, Mobility support for fog computing: An SDN approach, *IEEE Commun Mag*, **56(5)** (2018) 53–59.

7 Bilal K D, Munam A S, Huniya S & Adnan N, Fog computing based automated accident detection and emergency response system using Android smartphone, *Int Conf Emerg Technol*, (IEEE) 2018, 1–6.

8 Manoj M, Tilak A & Charles C T, Fog computing and the Internet of Things: A review, *IEEE Int Conf Cyber Security Cloud Comput*, (IEEE) 2021, 10–12.

9 Dastjerdi A V, Gupta H, Calheiros R N, Ghosh S K & Buyya R, Fog computing: Principles, architectures, and applications, in *Internet of Things* edited by R Buyya and A V Dastjerdi (Morgan Kaufmann) 2016, 61–75.

10 Dastjerdi A V & Buyya R, Fog computing: Helping the Internet of Things realize its potential, *Comput*, **49(8)** (2016) 112–116.

11 Hu P, Dhelim S, Ning H & Qiu T, Survey on fog computing: architecture, key technologies, applications and open issues, *J Netw Comput Appl*, **98** (2017) 27–42.

12 Ma K, Bagula A, Nyirenda C & Ajayi O, An IoT-based fog computing model, *Sens*, **19(12)** (2019) 2783.

13 Xiao Y & Krunz M, QoE and power efficiency tradeoff for fog computing networks with fog node cooperation, *IEEE Conf Comput Commun*, (IEEE) 2017, 1–9.

14 Atlam H F, Walters R J & Wills G B, Fog computing and the Internet of things: A review, *Big Data Cogn Comput*, **2(2)** (2018) 10.

15 Jamil B, Shojafar M, Ahmed I, Ullah A, Munir K & Ijaz H, A job scheduling algorithm for delay and performance optimization in fog computing, *Concurrency Comput Pract Exp*, **32(7)** (2020) e5581.

16 Harshit G, Amir V D, Soumya K G & Rajkumar B, iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, Edge and Fog computing environments, *Softw Pract Exp*, **47(9)** (2017) 1275–1296.

17 Mohammad M S & Maha K A, Collaborative execution of distributed mobile and IoT applications running at the edge, *Int Conf Electr Comput Technol Appl*, (IEEE) 2017, 1–5.

18 Fatema T Z, Md. R R K, Md. F R B & Amit K D, Enhancing the capabilities of IoT based fog and cloud infrastructures for time sensitive events, *Int Conf Elect Eng Comput Sci*, (IEEE) 2017, 224–230.

19 Abdullahi I, Arif S & Hassan S, Ubiquitous shift with information centric network caching using fog computing, in *Comput Intell Inf Syst* : *Proceedings of the Fourth INNS Symposia Series on Computational Intelligence in Information Systems* (Springer International Publishing) 2015, 327–335.

20 Ru H, Fei R Y, Tao H, Renchao X, Jiang L & Victor C M L, Software defined networking, caching, and computing for green wireless networks, *IEEE Commun Mag*, **54** (2016) 185–193.

21 Per-Olov Ö, James B, Paolo C, Philip E, Antonio F A & Johan F, Reliable capacity provisioning for distributed cloud/edge/fog computing applications, *Eur Conf Netw Commun*, (IEEE) 2017, 1–6.

22 Shree K S & Xianbin W, Live data analytics with collaborative edge and cloud processing in wireless IoT networks, *IEEE Access*, **5** (2017) 4621–4635.

23 Engin Z, Ejder B, Mehdi B, Manhal A K, Ilyas A K & Ahmet S E, Big data caching for networking: Moving from cloud to edge, *IEEE Commun Mag*, **54(9)** (2016) 36–42.

24 Talal H N, Sherali Z, Abdullah A & Quan Z S, Mobile cloud computing: Challenges and future research directions, *J Netw Comput Appl*, **115** (2018) 70–85.

25 Peter M C & Anders A, Emerging trends in electricity consumption for consumer ICT, *Proc Tech Rep* (National University of Ireland, Galway, Connacht, Ireland,) 2013, 1–56.

26 Kezhi W, Kun Y & Chathura S M, Joint energy minimization and resource allocation in C-RAN with mobile cloud, *IEEE Trans Cloud Comput*, **6(3)** (2018) 760–770.

27 Matías H, Juan M R, Cristian M & Alejandro Z, A two-phase energy-aware scheduling approach for CPU-intensive jobs in mobile grids, *J Grid Comput*, **15** (2017) 55–80.

28 Li C & Li L, Exploiting composition of mobile devices for maximizing user QoS under energy constraints in mobile grid, *Inf Sci*, **279** (2014) 654–670.

29 Weisong S, Jie C, Quan Z, Youhuizi L & Lanyu X, Edge computing: Vision and challenges, *IEEE Internet Things J*, **3(5)** (2017) 637–646.

30 Kai L, Liqiang Z, Xiaoli C & Hsiao-Hwa C, An integrated architecture for software defined and virtualized radio access networks with fog computing, *IEEE Netw*, **31(1) (**2017) 80–87.

31 Shao-Chou H, Hsiang H, Shao-Yu L & Kwang-Cheng C, Architecture harmonization between cloud radio access networks and fog networks, *IEEE Access*, **3** (2015) 3019–3034.

32 Sergej S, James B, Paul L, PJ B, Christian S, Henning G, Zafeirios P & Dimitrios N, Towards automated data-driven model creation for cloud computing simulation, *EAI Endorsed Trans Cloud Syst*, **2(5)** (2015) 248–255.

33 Rajkumar B, Rajiv R & Rodrigo N C, Modeling and simulation of scalable cloud computing environments and the CloudSim toolkit: Challenges and opportunities, *Int Conf High Perform Comput Simul*, (IEEE) 2009, 1–11.

34 Vasileios M, Panagiotis S & Ioannis M, Towards distributed data management in fog computing, *Wirel Commun Mob Comput*, **2018** (2018) 7597686.

35 Charith P, Yongrui Q, Julio C E, Stephan R-M & Athanasios V V, Fog computing for sustainable smart cities: A survey, *ACM Comput Surv*, **50(3)** (2017) 1–43.

36 Amir H A, Pengcheng J, William G B & Nizar L, Internet of Things-enabled smart cities: State-of-the-art and future trends, *Meas*, **129** (2018) 589–606.

37 Mung C & Tao Z, Fog and IoT: An overview of research opportunities, *IEEE Internet Things J*, **3(6)** (2016) 854–864.

38 Praveen K B & Hariharan K, Time series traffic flow prediction with hyper-parameter optimized ARIMA models for intelligent transportation system, *J Sci Ind Res*, **81(4)** (2022) 408–415.

39 Junali J J & Suresh C S, Mortality prediction of victims in road traffic accidents (RTAs) in India using opposite population SGO-DE based prediction model, *J Sci Ind Res*, **80(11)** (2021) 1001–1007.

40 Vincenzo I, David G & Carlo G, Enhancing smart city connectivity: A multi-metric CNN-LSTM beam forming based approach to optimize dynamic source routing in 6G networks for MANETs and VANETs, *Smart Cities*, **7** (2024) 3022–3054.