Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 47 No. 02 (2026)

Impact of Fog Computing as an Extension of Cloud Computing

Jasmeet Kaur ,Neha Agrawal

Department of Information Technology,

Shri G.S. Institute of Technology & Science, Indore

Abstract:- Fog Computing has emerged as a promising extension of Cloud Computing to address the inherent limitations of centralized cloud architectures, particularly in latency-sensitive and bandwidth-intensive applications. By bringing computation, storage, and networking resources closer to the data source at the network edge, fog computing significantly reduces latency, minimizes bandwidth consumption, and enhances real-time decision-making capabilities. This paradigm shift enables efficient processing of massive data generated by Internet of Things (IoT) devices, smart cities, autonomous vehicles, and industrial automation systems. The integration of fog and cloud computing creates a hierarchical architecture that leverages the strengths of both paradigms—low-latency processing at the edge and powerful computational capabilities in the cloud. This paper examines the key impacts of fog computing, including improved Quality of Service (QoS), enhanced security and privacy, energy efficiency, and scalability. It further discusses the challenges such as resource management, security concerns, and standardization issues. The study highlights how fog computing complements cloud infrastructure, paving the way for next-generation distributed computing environments in the era of 5G/6G and massive IoT deployment.

Keywords: *Fog Computing, Internet of Things (IoT), Cloud Computing, Edge Computing*

1. Introduction

The most advanced apps of today are usually built on top of cloud services, which allows them to take advantage of cost savings, user-friendliness, elastic scalability, and the appearance of limitless resources . Although cloud services offer these clear advantages, they also have a significant drawback: because cloud data centers are centralized, they are usually located far from end users, which causes excessive access latencies. This is adequate for many application domains, such online or enterprise apps, but not for more recent ones, like 5G mobile applications, Internet of Things (IoT)-based systems, or autonomous driving. As a result, edge devices are usually used for these applications. However, while such edge devices offer low access latencies due to their physical proximity to end users, they also come with limited resources and are subject to availability problems.[2]

In this case, using cloud services and edge nodes simultaneously is a clear way to get minimal latency and access to infinitely scalable resources. Fog computing is a paradigm that enables provisioning of resources and services at the network edge, closer to the end devices. It is both complementary to, and an extension of, the traditional cloud-based model, addressing several issues of connected devices, such as high bandwidth, geographical dispersion, and low latency needs. In addition to the advantages already mentioned, Fog Computing guarantees end users greater privacy: Future Fog apps can store sensitive personal data at the edge, sending only aggregated or appropriately anonymized data to the cloud, whereas present cloud applications gather personal data needs centrally.[2]

This paper aims to shed some light on Fog Computing and its role in service-oriented computing (SOC). To this aim, we firstly provide a definition of Fog Computing and the involved concepts relevant to SOC. Secondly, we identify the main obstacles for widespread fog adoption and describe a number of fundamental research challenges.[2]

1. Literature Review

Fog computing has emerged as a growing area of interest in both academic and industrial research. In recent years, numerous studies have been published on this computational paradigm, many of which take the form of literature reviews and surveys.

Several of these works have explored the role of computational resources within fog computing. One of the earliest contributions, introduced five types of fog nodes: servers, networking devices, cloudlets, base stations, and vehicles. In this work, fog nodes are described as "computational nodes with heterogeneous architecture and configurations capable of supporting Fog computing infrastructure at the network edge."[3] The authors further emphasize that a "Fog computing environment consists of traditional networking components such as routers, switches, set-top boxes, proxy servers, and base stations," which possess diverse capabilities in terms of computing, storage, and networking to facilitate the execution of service applications.[3]

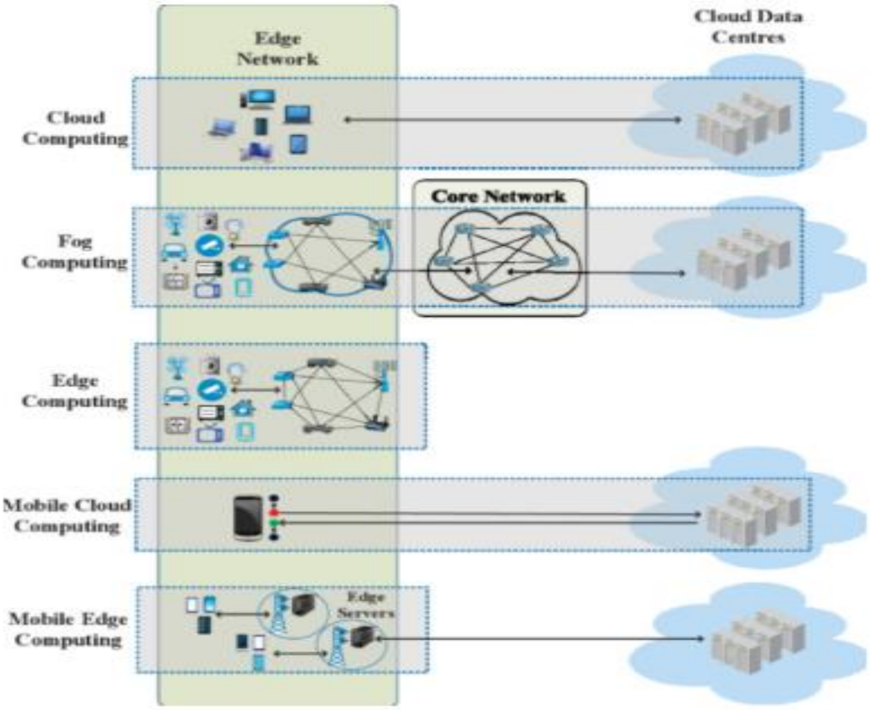


Fig. 1. Computation domain of cloud, fog, edge, mobile cloud and mobile edge computing [3]

Another study, proposed a taxonomy that classifies devices into three categories: IoT devices, processing devices, and gateway devices. This taxonomy aligns with the three layers of the fog computing architecture, as shown in Figure 1. In this framework, IoT devices correspond to the IoT layer, processing devices to the cloud layer, and gateway devices to the fog layer. Although the paper also outlined infrastructure (processing, storage, network, and memory) and network (connection and mobility) requirements, no formal definition of fog nodes was provided.

In , fog nodes are classified as either "smart"—those with some computational processing and storage capacity—or "dumb," referring to devices with limited computational capabilities, such as sensors and actuators. The authors define fog nodes as "distributed fog computing entities enabling the deployment of fog services, consisting of at least one or more physical devices with processing and sensing capabilities." They also highlight that "all physical devices within a fog node are connected by various network technologies, aggregated, and abstracted to form a single logical entity—i.e., the fog node—which can seamlessly execute distributed services as if they were on a single device."[4]

The study in presents a model for representing physical resources in a fog computing environment, focusing on processing, storage, memory, and networking capabilities. The authors also discuss the role of virtualization in fog computing due to its ability to handle heterogeneous systems; however, they only consider virtual machines and containers in their model, without offering a definition for fog nodes.[4]

Lastly, from a computational perspective, takes a broader approach, categorizing resources into hardware and software. Hardware resources include computing and network devices, while software resources encompass virtualization systems and virtualized networks. According to Hong and Varghese, devices in the fog environment must have both physical and virtual resources available to manage hardware effectively. However, no specific definition of fog nodes is provided by these authors.[4]

Table I compares the works discussed above with the objectives of this study. Notably, this paper is the only one that explores fog nodes from a computational perspective, covering both physical and virtual resources, including unikernels—a lightweight virtualization technique well-suited for fog computing environments.[4]

Criteria	**Cloud Computing**	**Fog Computing**	**Edge Computing**
Latency	High latency	Low latency	Very low latency
Capacity	No reduction in data while sending or transforming	Reduces the amount of data sent to the cloud	Processes data locally at edge devices, minimal data sent to fog/cloud
Responsiveness	Low responsiveness	High responsiveness	Highest responsiveness
Security	Less secure	More secure than cloud	Highest security, localized data processing
Speed	High access speed depending on VM connectivity	Faster than cloud	Fastest access speed
Data Integration	Multiple data sources can be integrated	Multiple data sources and devices can be integrated	Integrates devices at the edge
Mobility	Limited mobility	Supports mobility	Supports mobility
Location Awareness	Partially supported	Supported	Fully supported
Number of Server Nodes	Few server nodes	Millions of nodes	Billions of nodes
Geographical Distribution	Centralized	Decentralized	Highly decentralized, distributed across edge devices
Location of Service	Services provided within the internet	Services provided at the edge of the local network	Services provided directly at edge devices
Working Environment	Specific data center with air conditioning	Indoor or outdoor (houses, streets, cafes, base stations, etc.)	Edge devices, close to where data is generated

Communication Mode	IP network	Wireless (WiFi, 3G, 4G, ZigBee, etc.) and wired communication	Wireless communication or wired (usually local area networks)
Dependence on Core Network	Requires strong network core	Can work in weak network cores	Minimal reliance on network core, processes data locally
Scalability	Highly scalable	Less scalable than cloud, but more scalable than edge	Less scalable than fog
Node Location	Nodes installed far from the data source	Nodes closer to the cloud but farther from edge	Nodes installed at the edge, closest to the data source
Operational Cost	Comparatively lower	Moderate	Higher operational cost due to extensive edge device usage
Privacy	Higher probability of data attacks	More private than cloud	Very high privacy due to localized data processing
Power Consumption	High power consumption at centralized data centers	Moderate power consumption (nodes filter and forward data)	Low power consumption (data processed directly at edge devices)
Use Case	Suitable for general-purpose applications with less sensitivity to latency	Suitable for real-time applications requiring low latency and mobility support	Ideal for ultra-low latency applications, real-time processing (e.g., autonomous vehicles, IoT devices)

What is Fog Computing?

Fog Computing is a distributed computing paradigm, integrated into the cloud, whose processing is done at the network edge. It provides computing resources for applications that cannot perform properly with the high latency provided by cloud-only environments.

Bonomi et al. presented the first definition of fog computing stating that it is a highly virtualized platform that provides computing, storage, and networking services among many computing data centers or end-devices. These components may or may not be at the edge of the network.[5]

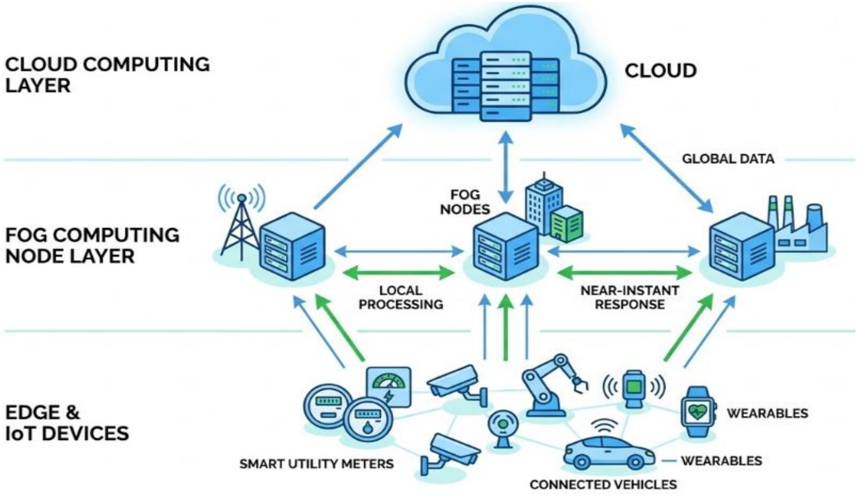


Figure 2: Architecure layer of Fog Computing

Several researchers have expanded and revised this initial definition of fog computing. Yi et al. consider fog computing as a scenario, composed of a high number of decentralized and heterogeneous devices, where they communicate and cooperate among themselves and with the network to perform data processing and storage without third-party interventions. Services, applications, or basic network functions that run in a sandboxed environment, can be supported by the data processing and storage.[5]

For Dastjerdi et al. [11], fog computing is considered a distributed computing paradigm. In this paradigm, the services provided by the cloud are essentially extended to the network edge. Fog computing addresses application requirements that need low latency with a huge and dense geographical distribution. Therefore, fog computing supports computing resources, different communication protocols, mobility, interface heterogeneity, integration with the cloud, and distributed data analytics.

For Naha et al.[12], fog computing is a distributed platform where the edge or end devices, that can be virtualized or not, will do the majority of processing. It resides in between the cloud and users and the cloud will do long-term storage and non-latency-dependent processing.

In the industry point of view, Cisco defines that the fog extends the cloud to be closer to the things that produce and act on Internet of Things (IoT) data. These devices, called fog nodes, can be any device with computing, storage, and network connectivity and are deployed anywhere with a network connection.

For IBM, the term “fog computing” and “edge computing” carry the same meaning. They both mean operation on network ends rather than hosting and working from a central cloud. They represent the scenario where processes and resources are located at the edge of the cloud and do not establish any channel for cloud utilization.[5]

The Open Fog Consortium, which since January 2019 has merged with the Industrial Internet Consortium, states that fog computing is a decentralized computing infrastructure that considers the best place between the cloud and the data source to distribute computation, storage, and applications. It is both complementary to, and an extension of, traditional cloud-based models. [5]

Finally, fog computing was defined by the National Institute of Standards and Technology (NIST) as a layered model facilitating the deployment of applications and services that are latency-aware and distributed. This model enables ubiquitous access to shared devices that are not perceptibly different from each other, although the extremes are quite distinct, from scalable computing resources. Therefore, fog computing provides, for the end-devices, local computing resources, network connectivity to centralized services and minimizes the requestresponse time, when needed.[5]

How can it be distinguished from Edge Computing?

Without mentioning cloud services, edge computing is just concerned with processing at the network's edge. Fog computing can be defined as the combination of cloud, edge, and any intermediate nodes (which could be small-to medium-sized data centers within the network provider's core network) or as the same as Edge Computing , depending on the source. We adopt the latter stance in this paper. For a high-level deployment overview of fog computing.[7]

In order for Fog Computing to reach its full potential, it must do more than just build Cloudlets, or data centers in a box, to move the cloud closer to data producers. Rather, fog computing needs to be viewed as a "resource layer that fits between the cloud data centers and the edge devices, with features that may resemble either."

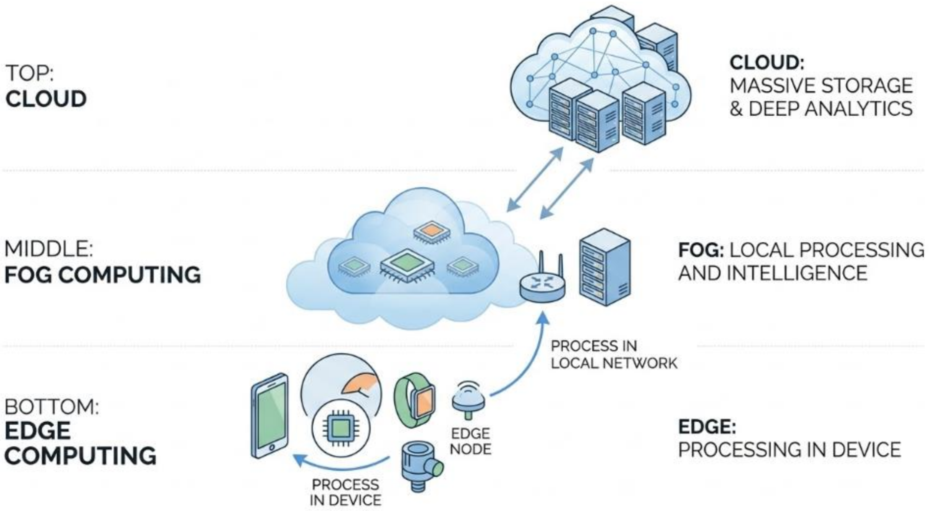


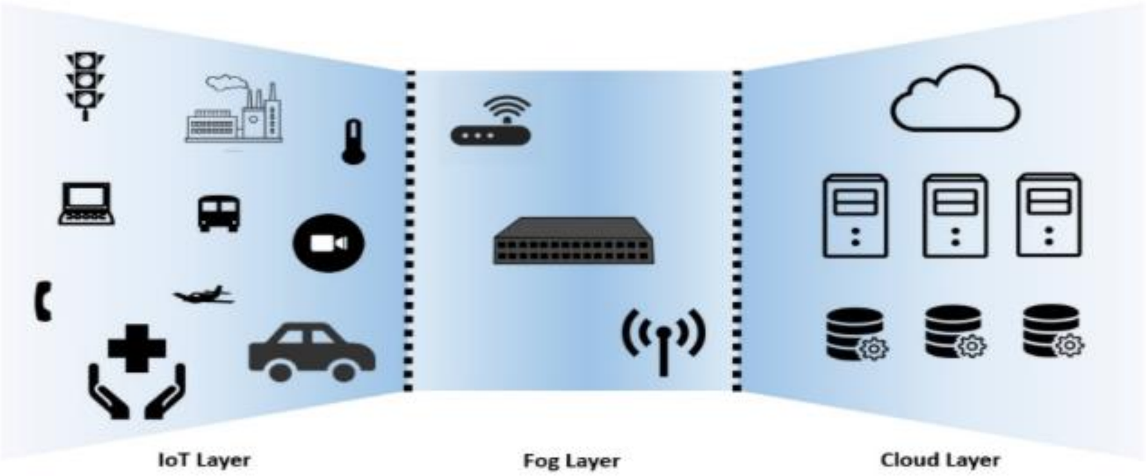
Figure 3: Show Edge computing architecture

The objective of Fog Computing is to offer a collection of techniques and resources to establish a continuum between edge and cloud, as well noted by the OpenFog Consortium . Fog computing technologies in particular must allow for smooth data flow in both directions between cloud services and the edge while meeting application quality of service (QoS) requirements. Such data transfer could also be accompanied by movement of computation – both in the same way and in a complimentary fashion compared to data movement.[7]

Architecture

The most popular method is the layered [or hierarchical] representation. The typical illustration of a fog computing environment in this context is a three-tiered architecture.

Nevertheless, plans with four , five, or even six layers can also be found. provides a thorough analysis of fog computing designs. Thus, as shown in Figure 1, a fog computing architecture consisting of three layers—the IoT Layer, the Fog Layer, and the Cloud Layer—will be employed in this work.[5] This study employs a horizontal architecture as opposed to other approaches that use hierarchical systems. The goal is to demonstrate that the cloud layer and the Internet of Things layer, which are located on Figure 1's left and right edges, respectively, have anumber of devices and resources that tend to the infinite, while the internal Fog layer has a more limited number of devices and computational resources.[8]



All of the IoT devices linked to the network's edge are represented by the IoT layer. End users submit service requests in this layer, which are then handled by the Fog and Cloud levels. Before sending the data to the cloud, the fog layer serves as a bridge between the cloud and IoT layers, offering the features required for data processing applications such aggregation and filtering . In addition to processing and storing data, this layer is made up of nodes, sometimes known as fog nodes, which are "smart" devices that may also route and forward data packets[8]to the cloud layer. Lastly, the Cloud layer has more powerful processing capabilities to handle all of the IoT layer's demands.[8]

Essential Characteristics

Regarding the characteristics of fog computing, although some publications are divergent and the ones most used by academics are those indicated by the NIST , described bellow:

- **Low Latency**: as fog computing devices are often located at the edge of the network, the analysis and response to data generated by these devices is much faster than from a cloud service;

- **Geographic Distribution**: in contrast to cloud computing, services and applications driven by fog computing require widely geo-distributed deployments;

- **Heterogeneity**: supports the collection and processing of data originating from different sources and acquired through various types of network communication resources;

- **Interoperability and Federation**: components must be able to interoperate and services must be federated between domains;

- **Real-Time Interactions**: fog computing applications involve real-time interactions, rather than batch processing;

- **Scalability**: supports scalability of computational resources, changes in workloads and variations in network and device conditions. Considering these characteristics, fog computing is suitable to use when cloud computing does not provide the latency or runtime requirements required by applications.[9]

The main obstacles for adoption of Fog Computing

Broadly, obstacles for a wide adoption of Fog Computing can be categorized as *inherent*, e.g., available technology or physical constraints, and *external* obstacles, e.g., privacy legislation. In this section, we will give an overview of both.

Inherent Obstacles

Several inherent challenges arise from the concept of using fog resources. These can include technical limitations, such as constrained computational power; logical constraints, like tradeoffs in distributed systems; and market challenges, such as the current absence of managed edge services.

- **O1: Absence of Edge Services**: While cloud services offer the convenience of a service-based consumption model, there are currently no edge infrastructure services that allow on-demand provisioning of compute or storage capacity. We expect managed edge services to emerge in the market eventually, potentially through partnerships between cloud providers and network carriers. Amazon is already making strides in this direction with its Greengrass service, which provides software for edge devices. However, since there is no established method to “rent” edge capacity on demand, fog application providers must currently build, manage, and maintain their own physical edge hardware, along with integrating it with cloud services.[10]

- **O2: Lack of Standardized Hardware**: Although some off-the-shelf edge devices are available, they come in various configurations, particularly in terms of compute power. Alternatives, such as Raspberry Pis, BeagleBoards, or custom-built solutions, further diversify the hardware landscape. This diversity leads to two main issues: first, software stacks must be adapted for each type of hardware and may not run universally across devices. Systems designed to be hardware-agnostic often fail to fully utilize the specific capabilities of available resources. Second, application architectures must be flexible enough to handle varying resource capacities, requiring highly modular applications that can deliver different features based on the capabilities of the devices they are running on.[10]

- **O3: Significant Management Effort**: Fog computing is primarily about bringing data and computation closer to end users, which results in large-scale deployments involving numerous fog nodes. Each of these nodes needs to be managed, whether for scaling, updates, or configuration changes. Since no managed fog infrastructure service exists, the burden of this management falls entirely on application providers. Compared to managing smaller on-premise data centers, this represents a substantial challenge.[11]

- **O4: Managing Quality of Service (QoS)**: As systems grow in size and geographic spread, they become more complex. This increase in complexity introduces more frequent issues, such as network latency, message reordering, and loss, as well as more serious problems like network partitioning or Byzantine failures. While systems can generally handle these challenges, they still impact quality of service and necessitate trade-offs. In large-scale geo-distributed systems, these issues occur more frequently simply due to the increased number of nodes. Moreover, fog applications in areas like IoT or autonomous driving impose stricter demands on infrastructure reliability since the consequences of failure are far more severe. For example, while a fault in a social network might result in a lost private message, the same fault in an autonomous vehicle could cause injury or even death.[11]

- **O5: Lack of Network Transparency**: Distributed systems have traditionally sought to hide the complexities of distribution, allowing applications to function as though they were running on a single machine. Over time, certain aspects of this transparency have been sacrificed in favor of other QoS goals. One aspect that remains largely hidden is network topology. Cloud services, for instance, provide only basic high-level details about their distribution (e.g., availability zones in AWS) and rely heavily on network virtualization. In fog computing, however, the interconnection of nodes must account for the underlying network hierarchy. For example, two edge nodes in close geographical proximity may be connected directly or through several layers of network hierarchy, such as the Internet backbone. This results in varying degrees of latency between nodes. Therefore, fog applications can no longer rely on broad abstractions, such as an AWS region, and must instead be aware of actual network topologies to avoid unexplained performance fluctuations. This highlights the need for new, finer-grained topology abstractions to improve manageability within fog applications.[12]

Conclusion

Fog computing has emerged as a transformative extension of traditional cloud computing, effectively bridging the gap between centralized data centers and the rapidly growing number of edge devices in the Internet of Things (IoT) era. By bringing computation, storage, and networking resources closer to the data source, fog computing significantly reduces latency, minimizes bandwidth consumption, and enables real-time decision-making—capabilities that pure cloud architectures often struggle to deliver for latency-sensitive applications such as autonomous vehicles, smart healthcare, industrial automation, and augmented reality.

The impact of fog computing is profound. It complements rather than replaces cloud computing by offloading time-critical tasks to intermediate fog nodes while reserving the cloud for heavy-duty, long-term analytics, storage, and global-scale processing. This hybrid approach enhances overall system efficiency, improves data privacy and security through localized processing, and lowers operational costs by reducing unnecessary data transmission to distant cloud servers. Furthermore, fog computing addresses key limitations of cloud computing, including network congestion, high latency, and dependency on constant internet connectivity.Despite its advantages, challenges such as resource heterogeneity, security vulnerabilities at the edge, complex orchestration, and standardization issues remain areas requiring continued research and innovation. Looking ahead, the convergence of fog, edge, and cloud computing will play a pivotal role in shaping the future of intelligent systems. As 5G/6G networks proliferate and IoT adoption accelerates, organizations that strategically integrate fog computing into their architectures will gain a competitive edge through enhanced agility, reliability, and responsiveness. In conclusion, fog computing represents not just an incremental improvement but a paradigm shift that extends the horizon of cloud computing, enabling a more distributed, efficient, and intelligent computing ecosystem for the digital future.

References

1. Das, R. "A Review on Fog Computing: Issues, Characteristics, and Trends." *ScienceDirect*, 2023
2. Habibi, P. "Fog Computing: A Comprehensive Architectural Survey." *IEEE Xplore*, 2020
3. Dhanaraj, R. K. "A Review Paper on Fog Computing Paradigm to Solve Cloud Computing Issues in IoT Devices." *IOP Science*, 2021
4. Ahammad, I. "Fog Computing Complete Review: Concepts, Trends, and Applications." *Springer*, 2023
5. Adel, A. "Utilizing Technologies of Fog Computing in Educational IoT." *SpringerOpen*, 2020
6. Puliafito, Claudio, et al. "Fog Computing for the Internet of Things: A Survey." *ACM Digital Library*, 2019
7. Atlam, Hany F., et al. "Fog Computing and the Internet of Things: A Review." *MDPI*, 2018
8. Srirama, Satish Narayana, et al. "A Decade of Research in Fog Computing." *arXiv*, 2023
9. Top 15+ Interesting Fog Computing Projects." *PhD Services*
10. Mahmud, R., et al. "Fog Computing: A Taxonomy, Survey and Future Directions." *Download Paper*, 2018
11. A. V. Dastjerdi *et al.*, "Distributed intrusion detection in clouds using mobile agents," *3rd International Conference on Advanced Engineering Computing*, 2009.
12. M. Puri, "A response to Naha et al.," *ScienceDirect*, 2021
13. "Fog Computing: An Overview." *International Journal of Trend in Research and Development*
14. Rathore, P., Gupta, P., Jain, S. and Shrivastava, Y., 2022. A Study of the Automated Vehicle Number Plate Recognition System. i-manager's Journal on Pattern Recognition, 9(2), p.30.
15. Gupta, P., Sharma, V. and Varma, S., 2022, September. An Algorithm for Counting People using Dense Nets and Feature Fusion. In 2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA) (pp. 1248-1253). IEEE.
16. Gupta, P., Varma, S., Arya, N. and Bhagel, R., 2023. Intelligent Security System Based on the Internet of Things (IoT). In Intelligent Sensor Node-Based Systems (pp. 177-191). Apple Academic Press
17. Kushwaha, U., Gupta, P., Airen, S. and Kuliha, M., 2022, December. Analysis of CNN Model with Traditional Approach and Cloud AI based AppGupta, P., Sharma, V. and Varma, S., 2022. A novel algorithm for mask detection and recognizing actions of human. Expert Systems with Applications, p.116823.roach. In 2022 International Conference on Automation, Computing and Renewable Systems (ICACRS) (pp. 835-842). IEEE.
18. Chaturvedi, A. and Gupta, P., 2021. The Cloud: Features, Challenges and Scope. International Journal of Progressive Research in Science and Engineering, 2(8), pp.466-471.
19. Gupta, P., Sharma, V. and Varma, S., 2021. People detection and counting using YOLOv3 and SSD models. Materials Today: Proceedings.
20. Gupta, P., Kulkarni, A. and Sarda, A., 2013. An embedded health care supervisory systems. International Journal of Latest Trends in Engineering and Technology (IJLTET), 3, pp.379-386
21. Gupta, P., Shukla, M., Arya, N., Singh, U. and Mishra, K., 2022. Let the Blind See: An AIIoT-Based Device for Real-Time Object Recognition with the Voice Conversion. In Machine Learning for Critical Internet of Medical Things (pp. 177-198). Springer, Cham.
22. Gupta, P., Arya, N., Singar, C.P., Chaudhari, A., Singh, U. and Gupta, S., 2025. Safety of Pedestrians in AI-Optimized VANETs for Autonomous Vehicles via Real-Time Vehicle-to-Vehicle Communication. In AI-

Driven Transportation Systems: Real-Time Applications and Related Technologies (pp. 169-181). Cham: Springer Nature Switzerland.

23. Manurkar, V., Gupta, P., Sharma, G., Singh, U. and Manurkar, N., 2022. A Face Mask Identification System based on the Internet of Things and Machine Learning for Detecting Covid-19. NeuroQuantology, 20(16), p.3930.