World Journal of Engineering Research and Technology

www.wjert.org

Impact Factor: 7.029 Coden USA: WJERA4

THE ROLE OF EDGE COMPUTING IN REAL-TIME IOT DATA PROCESSING

***[1]Mr. Anik Bhattacharya, [2]Dr. Mohammed Bakhtawar Ahmed**

*[1]Student, KK Modi University.

[2]Head of Department, KK Modi University.

Article Received on 18/12/2025 Article Revised on 07/01/2026 Article Published on 01/02/2026

***Corresponding Author**

Mr. Anik Bhattacharya

Student, KK Modi University.

https://doi.org/10.5281/zenodo.18440975

How to cite this Article: *[1]Mr. Anik Bhattacharya, [2]Dr. Mohammed Bakhtawar Ahmed. (2026). THE ROLE OF EDGE COMPUTING IN REAL-TIME IOT DATA PROCESSING. World Journal of Engineering Research and Technology, 12(2), 11–18.

ABSTRACT

With the rapid growth of IoT devices, traditional cloud computing faces challenges such as high latency and bandwidth overload, making it unsuitable for real-time applications like industrial control and smart transportation. This paper introduces a real-time IoT data processing approach based on edge computing, enabling efficient local processing through layered architecture design, core technology integration, and optimized workflows. In industrial IoT scenarios, the method reduces equipment downtime from 12 to 4.8 hours per month and shortens data processing delay from 200–300 ms to just 10 ms. In smart city applications, it decreases average travel time by 20% and lowers traffic congestion by 18%. For agricultural IoT, it cuts irrigation water use by 35% while increasing crop yields by 15%. Overall, the proposed method reduces processing delays by 60–95%, improves system throughput by 35%, and enhances resource utilization—offering a reliable, high-performance solution for real-time IoT data processing.

INTRODUCTION

The Internet of Things (IoT) is rapidly transforming every sector, from smart homes and cities to industrial automation and intelligent transportation. According to IDC, by 2025 there will be over 41.6 billion connected devices generating nearly 79 zettabytes of data daily. Traditional cloud computing struggles to handle such massive real-time data due to high latency, bandwidth consumption, and transmission costs. Edge computing addresses these

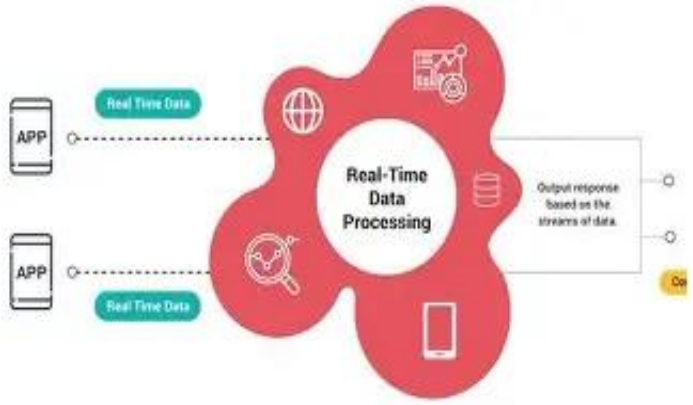
challenges by processing data closer to its source, significantly reducing response time, network load, and operational costs. It enables faster decision-making, ensures reliability even during network disruptions, and enhances data security and privacy. Thus, edge computing plays a crucial role in enabling efficient, secure, and real-time IoT data processing for modern intelligent systems.

Fundamentals of Edge Computing

Edge computing is a distributed computing paradigm that processes data closer to its source rather than relying entirely on centralized cloud infrastructures. It brings computation and storage resources to the network's edge—near IoT devices, sensors, and gateways—where data is generated. This proximity enables faster processing, reduced latency, and improved system efficiency. The core principles of edge computing revolve around decentralization, scalability, and efficiency in handling large volumes of real-time data.

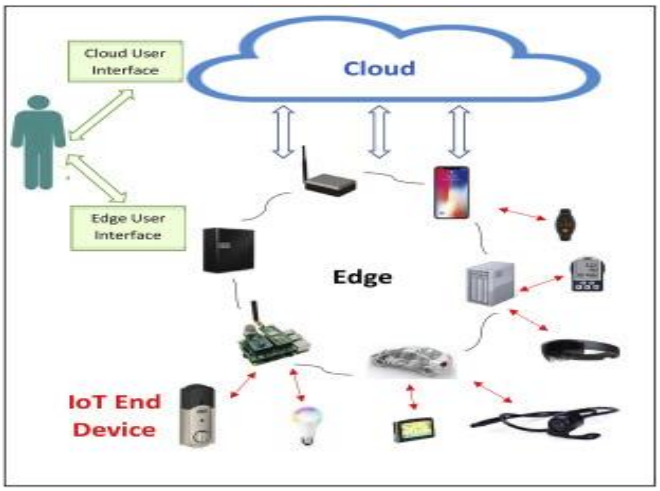
The architecture of edge computing typically includes multiple components such as edge nodes, gateways, and micro data centres. These components collaborate to process, analyse, and store data locally before transmitting only relevant or summarized information to the cloud. This distributed framework enhances system performance, reliability, and security by minimizing data transmission delays and dependency on distant data centres. It also allows edge devices to continue functioning independently during network disruptions, ensuring uninterrupted operations. Compared to traditional cloud-centric models, edge computing offers significant advantages. By processing data locally, it dramatically reduces latency, making it ideal for time-sensitive applications such as autonomous vehicles, industrial automation, and healthcare monitoring. It strengthens data privacy and security by keeping sensitive information within localized networks, reducing exposure to potential cyber threats. Furthermore, by filtering and aggregating data before transmission, edge computing

decreases bandwidth usage and operational costs. In real-time IoT data processing, edge computing enhances decision-making and responsiveness. Its scalable and adaptive architecture supports fluctuating workloads, ensuring optimal performance as the number of connected devices grows. Overall, edge computing provides an efficient, secure, and resilient foundation for the next generation of intelligent and real-time applications.



Real-Time Data Processing at the Edge

Edge computing leverages distributed computational resources deployed across the network's edge to perform real-time data processing tasks. These resources include edge servers, IoT devices, and gateways, which collectively enable efficient computation and analysis of data closer to its source. By processing data locally at the edge, latency issues inherent in centralized processing models are significantly mitigated. This proximity to data sources reduces the time required for data transmission, resulting in faster response times and improved overall system performance. Real-time data processing at the edge enables enhanced responsiveness for critical applications, such as IoT deployments, autonomous vehicles, and industrial automation. By minimizing the delay between data generation and action, edge computing ensures timely responses to events, improving user experience and operational efficiency. Edge computing facilitates the implementation of real-time analytics capabilities by analysing data streams locally and deriving actionable insights in near real-time. This enables organizations to make informed decisions quickly, based on up-to-date information, without relying on centralized data processing infrastructure. Additionally, edge analytics can reduce the volume of data transmitted to centralized systems, optimizing network bandwidth and storage.



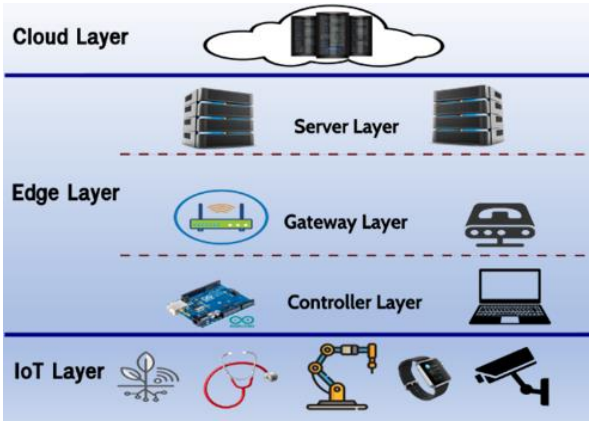
Edge-based security designs for IoT

With the emergence of edge computing in recent years, many researchers have explored edge computing-based designs to tackle the IoT security challenges. These designs range from comprehensive security architecture designs to specific designs to achieve dedicated security goals, such as distributed firewalls, intrusion detection systems, authentication and authorization algorithms, and privacy-preserving mechanisms. In this section, we summarize the proposed designs and discuss their strengths and weaknesses, respectively. Because there are only a small number of edge-based IoT security solutions, we try our best to include all related and quality work in this survey.

- The edge provides a new venue to design and deploy novel and comprehensive security solutions for IoT applications. These designs target at satisfying most security needs of end devices by maximally offloading security protection from end devices to the edge layer. Placing security mechanisms at a trusted edge layer can alleviate the security challenges caused by resource constraints at the IoT device layer.
- User-centric edge-based IoT security architecture. User satisfaction is one of the key factors in the success of IoT applications. With billions of IoT devices connected to an Internet-scale network, IoT applications provide users chances to access an enormous amount of resources in the system through various terminals, such as PCs, smartphones, smart TVs, smartwatches, etc.
- The convenience and pervasive resource accessibility are the most attractive features of IoT applications. However, when security is considered, there are two significant concerns. On the one hand, the user may not login in from an always trusted and secured device. On the other, most regular users may not have enough knowledge of effectively managing security. Therefore, it is risky to rely on users for security. Having the edge

layer managing the security for each specific user becomes an attractive idea, which results in security architecture designs such as offloading personal security to the network edge and virtualized security at the network edge.

Architecture of Edge Computing-Based IoT



In the context of IoT, edge computing is primarily focused on its implementation across various IoT scenarios, aiming to minimize decision-making latency and network traffic. The edge computing-based IoT architecture consists of three distinct layers: IoT, edge, and cloud, all of these are built on top of existing edge computing reference designs. Our primary focus is to define the specific functions allocated to each layer and explore the communication mechanisms established among these layers.

- **IoT layer**: The IoT layer encompasses a broad spectrum of devices and equipment, such as smart cars, robots, smart machinery, handheld terminals, instruments and meters, and other physical goods. These objects are tasked with overseeing the functioning of services, activities, or equipment. Furthermore, The IoT layer consists of actuators, sensors, controllers, and gateways constructed expressly for IoT contexts, which enable the administration of computational resources within IoT devices.

Edge layer: The main purpose of this layer is to receive, process, and send streams of data from the Device Layer. It offers real-time services like intelligent computing, security and privacy protection, and data analysis. Based on the equipment's ability to process data, three further sub-layers are separated from the Edge Layer: the near-edge layer, the mid-edge layer, and the far-edge layer.

- Far-Edge Layer (Edge controller layer): In this layer, data is collected from the IoT layer by edge controllers and subsequently undergoes initial threshold assessment or data filtering. After that, the edge layer or cloud layer directs the control flow back to the IoT

layer. After IoT device data has been collected, it is pre-processed to determine thresholds or perform data filtering. Consequently, the edge controllers in this layer must incorporate algorithm libraries tailored to the environment's configuration to consistently improve the strategy's efficiency. Additionally, these edge controllers should convey the control flow back to the IoT layer via the programmable logic controller (PLC) control or action control module after receiving decisions from the edge controller layer or upper layers.

- Mid-Edge Layer (Edge gateway layer): This layer is often made up of edge gateways, which can connect to wired networks like industrial ethernet or wireless networks like 5G to receive data from the edge controller layer. Furthermore, the layer enables diverse processing capabilities and caches the accumulated data. Moreover, the edge gateways in this layer play a crucial role in shifting control from the upper layers, such as the cloud layer or edge server layer, to the edge controller layer. Simultaneously, they monitor the equipment in both the edge gateway layer and the edge controller layer. The mid-edge layer has more storage and processing power than the far-edge layer, which can only carry out basic threshold judgment or data filtering. As a result, it can handle IoT layer data in a more thorough manner.
- Near-Edge Layer (Edge server layer): The edge server layer is equipped with robust edge servers. Within this layer, advanced and crucial data processing takes place. The edge servers leverage dedicated networks to gather data from the edge gateway layer and generate directional decision instructions based on this collected information. Additionally, platform administration and business application management features are anticipated for the edge servers in the edge server layer.
- **Cloud layer**: This layer primarily focuses on in-depth data mining and seeks to allocate resources optimally on a big scale, across a whole organization, a region, or even the entire country. Data from the edge layer is sent to the cloud layer through the use of the public network. Additionally, the edge layer has the ability to receive feedback from cloud layer-provided business applications, services, and model implementations.

Implementation patterns & technologies

- **Lightweight stream processing** - Streaming ETL pipelines—filtering, aggregation, feature extraction—reduce raw data volumes and perform initial analytics at the edge. Containerization and message buses (e.g., ZeroMQ) allow portable and modular edge services, enabling faster deployment and easier management across heterogeneous hardware.

- **Edge AI and model optimization** - Techniques such as model quantization, pruning, and knowledge distillation allow neural networks to run on edge accelerators (e.g., Jetson, Coral). For many real-time use cases, inference at the edge is feasible and significantly reduces latency versus cloud inference.
- **Protocols & communication patterns** - MQTT, CoAP, and lightweight gRPC variants are commonly used; design choices aim to minimize jitter and enable QoS guarantees. Real-time systems often use prioritized, event-driven messaging and local control loops to guarantee responsiveness.

Benefits for real-time IoT workloads

- Lower latency: Edge processing reduces end-to-end response time by eliminating long-haul trips to the cloud—critical for tactile and safety-critical systems.
- Bandwidth savings: Preprocessing and selective forwarding at the edge reduce upstream data volumes and operating costs.
- Improved privacy: Local processing can anonymize or filter sensitive data before transmission.
- Resilience: Edge nodes can operate when connectivity is intermittent, continuing local control and alerts.

Challenges and open research problems

- **Resource heterogeneity and constrained devices** - Edge nodes range from tiny microcontrollers to powerful micro-data centers. Scheduling and adapting algorithms across this spectrum (so real-time constraints are met) remain hard problems.
- **Orchestration & lifecycle management** - Managing deployments, updates, and migrations across many distributed nodes while maintaining real-time guarantees is an open systems challenge—especially when nodes are mobile or operate in unreliable networks.
- **Security at scale** - Distributing processing increases the number of endpoints to defend; lightweight yet robust authentication, secure boot, and anomaly detection at the edge are active research areas. Federated approaches help but introduce new attack vectors and privacy trade-offs.
- **Predictable real-time guarantees** - Providing provable, deterministic latency bounds in the face of resource contention, network variability, and dynamic workloads is still

largely unsolved beyond carefully bounded industrial settings. Research into real-time OS support, network slicing (5G), and priority scheduling is ongoing.

Future directions

- Edge-native ML toolchains: Integrated toolchains that automatically shrink and deploy models to appropriate edge tiers while verifying latency constraints.
- Standardized edge orchestration: Cross-vendor, interoperable orchestration and observability stacks that make it simpler to guarantee SLAs at the edge.
- Secure federated architectures: Federated learning with stronger privacy guarantees and robust aggregation techniques for distributed model updates.
- Edge-cloud co-design: Joint optimization algorithms that partition workloads across device/edge/cloud based on latency, cost, and energy models in real time.

CONCLUSION

Edge computing is a practical and increasingly mature approach for processing IoT data that demands real-time response. Evidence from surveys and system prototypes shows measurable latency, bandwidth, privacy, and resilience benefits when architectures and software are designed for the edge. However, to fully realize real-time guarantees at scale, further work is needed on orchestration, secure and lightweight analytics, and predictable performance across heterogeneous edge fleets. The research directions above offer a path for closing these gaps and making edge-first real-time IoT systems robust and widely deployable.

REFERENCES

1. Andriulo, F. C., et al. *Edge Computing and Cloud Computing for Internet of Things: A Review. Humanities and Social Sciences Communications* / MDPI, 2024.
2. *Edge Computing for IoT* (arXiv preprint), Feb 20, 2024 — chapter and survey of edge paradigms, layered architectures, and edge intelligence.
3. Sha, K., Yang, T. A., Wei, W., Davari, S. *A Survey of Edge Computing-based Designs for IoT Security,* 2020 (survey).
4. Urblik, L., et al. *A Modular Framework for Data Processing at the Edge: Design and Implementation. Sensors* (MDPI), 2023 — containerized, streaming ETL at the edge.
5. Truong T. H., Ta P. B., Dao M. L., et al. *LocKedge: Low-Complexity Cyberattack Detection in IoT Edge Computing.* arXiv: 2011.14194, 2020 — edge-level anomaly detection prototype.