Journal of Internet of Things and Information Technology Things
Volume 9, Issue 1, January-April 2026
ISSN: 3107-4502 (Online)

Edge Computing for IoT Systems: Architecture, Applications, Challenges and Future Directions

Sabita Sharma[1], Devansh Kulkarni[2], Megha Patil[3], Sambhu Rawat[4]
Associate Professor, Students
Department of Computer Science Engineering
Shantiniketan Institute of Polytechnic
Email id: *sabitasharm10a@gmail.com[1],kulkarnidevansh31@rediffmail.com,[2]*
meghapatil04@yahoo.com[3]
DOI: *https//doi.org/10.5281/zenodo.19275911*

ABSTRACT

Internet of Things (IoT) systems are generating very large amount of data from sensors, devices and smart objects. Traditional cloud computing model is not always suitable for IoT applications because of latency, bandwidth consumption and privacy concerns. Edge computing is emerging as an effective solution by processing data near to the source rather than sending everything to distant cloud servers. This paper presents a detailed review of edge computing for IoT systems including its architecture, enabling technologies, applications, benefits and challenges. Different edge architectures such as fog computing, mist computing and cloudlets are discussed. Role of edge computing in smart cities, healthcare, agriculture, industrial IoT and transportation is also explained. The paper also includes comparison tables and figures to understand the concepts better. Finally, challenges and future research directions are presented. Edge computing is expected to play very important role in next generation IoT ecosystem.

KEYWORDS*: Edge computing, IoT, fog computing, latency, smart devices, data processing, real-time systems.*

INTRODUCTION

IoT is connecting billions of devices like sensors, cameras, actuators and smart objects to internet. These devices continuously generate data which is traditionally sent to centralized cloud for storage and processing. But this approach causes high delay, heavy network traffic and sometimes security risk. Many IoT applications such as autonomous vehicles, healthcare monitoring and industrial automation require real-time response, which cloud alone cannot provide.

Edge computing shifts computation from cloud to the edge of network, closer to devices. This reduces latency and improves efficiency. Edge nodes can be routers, gateways, local servers or even smart devices itself. With edge computing, only important data is sent to cloud while rest is processed locally.

This paper reviews how edge computing supports IoT systems and why it is becoming necessary.

NEED OF EDGE COMPUTING IN IOT

Traditional IoT architecture mostly depends on cloud computing where all sensor data is transmitted to centralized servers for processing and storage. While cloud offers high computational power and scalability, it is not always suitable for time-critical and data-intensive IoT applications. As number of connected devices is increasing rapidly, the limitations of cloud-centric model are becoming more visible. Edge computing is introduced to overcome these limitations by bringing computation closer to data source.

The following issues explain why edge computing is highly needed in IoT systems.

1. High Latency in Data Transmission

In cloud-based IoT systems, data generated from sensors must travel long distance through internet to reach cloud servers. After processing, response is again sent back to device. This round-trip time causes delay which is unacceptable for real-time applications.

FOR EXAMPLE:

- Autonomous vehicles require decision in milliseconds
- Health monitoring devices must alert immediately
- Industrial machines need instant fault detection

Such applications cannot wait for cloud response. Edge computing processes data locally at gateway or nearby server, reducing latency drastically and enabling real-time action.

2. Bandwidth Overload Due to Massive Data

IoT devices like cameras, sensors, smart meters generate huge volume of data continuously. Sending all raw data to cloud consumes very high network bandwidth.

EXAMPLE:

- CCTV cameras stream HD video 24×7
- Environmental sensors send readings every second
- Smart city devices produce terabytes of data daily

This leads to network congestion and increased cost. With edge computing, data is filtered, aggregated and pre-processed locally. Only important or summarized data is forwarded to cloud, which reduces bandwidth usage significantly.

3. Privacy Issues When Sending Sensitive Data to Cloud

Many IoT applications deal with sensitive data such as:

- Patient health records
- Personal home surveillance data
- Industrial confidential data
- Location and identity information

Transmitting such data to remote cloud increases risk of data breach, unauthorized access and cyber attacks. Edge computing allows data to be processed locally without exposing raw information to internet. This improves privacy and data security.

4. Dependency on Continuous Internet Connectivity

Cloud-based IoT systems depend heavily on stable internet connection. If connectivity is lost due to network failure or remote location, system may stop functioning.

Examples:

- Agricultural fields in rural areas
- Remote oil and gas pipelines
- Vehicles in tunnels or remote highways

Edge computing enables local processing even without internet. Devices can continue to operate and synchronize with cloud when connection is restored.

5. Higher Operational Cost

Sending large volume of data to cloud requires high storage, computing resources and data transfer cost. Organizations have to pay for:

- Cloud storage charges
- Data transfer fees
- Processing charges
- High bandwidth internet

By processing data at edge, only necessary data is stored in cloud which reduces operational cost.

6. Scalability Issues with Growing IoT Devices

As number of IoT devices increases, cloud servers face heavy load. Managing millions of devices centrally becomes complex. Edge computing distributes the load among multiple edge nodes, making system more scalable and efficient.

7. Reliability and Fault Tolerance

If cloud server fails or network is down, entire IoT system may be affected. Edge computing provides local decision making capability which increases system reliability and fault tolerance.

Edge computing solves these issues by providing local data processing.

Table 1: Cloud vs Edge Computing for IoT

Parameter	Cloud Computing	Edge Computing
Latency	High	Very Low
Bandwidth usage	High	Reduced
Data processing	Centralized	Distributed

Parameter	Cloud Computing	Edge Computing
Privacy	Less secure	More secure
Real-time response	Difficult	Possible
Internet dependency	Required	Not always required

EDGE COMPUTING ARCHITECTURE

Edge computing architecture is designed in a layered manner so that data can be processed at different levels depending on urgency, size and importance. Instead of sending every data packet to cloud, the architecture allows intelligent distribution of computation across device, edge and cloud. This layered design improves performance, reduces delay and optimizes resource usage.

The architecture mainly consists of three important layers:

1. Device Layer – Sensors and IoT Devices

This is the lowest layer of the architecture where actual data is generated. It includes:

- Sensors (temperature, humidity, motion, vibration, etc.)
- Cameras and microphones
- Wearable devices
- Smart appliances
- Industrial machines with embedded sensors

These devices continuously sense environment and produce raw data. However, they usually have limited processing power, memory and battery life. Therefore, they are not suitable for heavy computation but can perform very basic preprocessing like signal conditioning or simple filtering.

Functions at Device Layer:

- Data sensing and collection
- Basic signal preprocessing
- Communication with edge nodes through protocols like MQTT, CoAP, ZigBee, BLE, Wi-Fi

2. Edge Layer – Gateways, Routers and Edge Servers

This is the most important layer in edge computing architecture. It is located very close to IoT devices and acts as an intermediate processing unit between devices and cloud.

Edge layer components include:

- IoT gateways
- Local servers
- Routers with computing capability
- Micro data centers
- Base stations in 5G networks

These nodes have higher computational capability compared to devices. They perform tasks such as data filtering, aggregation, analytics and decision making.

Functions at Edge Layer:

- Data filtering and aggregation
- Real-time analytics
- Local decision making
- Temporary data storage
- Security enforcement and encryption
- Communication with cloud

For example, in smart surveillance system, video feed is processed at edge to detect suspicious activity. Only alerts or snapshots are sent to cloud instead of full video stream.

3. Cloud Layer – Long Term Storage and Heavy Analytics

The cloud layer is located far from devices in centralized data centers. It has very high storage and computational power.

Functions at Cloud Layer:

- Long-term data storage
- Historical data analysis

Journal of Internet of Things and Information Technology Things
Volume 9, Issue 1, January-April 2026
ISSN: 3107-4502 (Online)

- Machine learning model training
- Large scale data analytics
- System-wide monitoring and control

Cloud is still important but it handles tasks that are not time-critical.

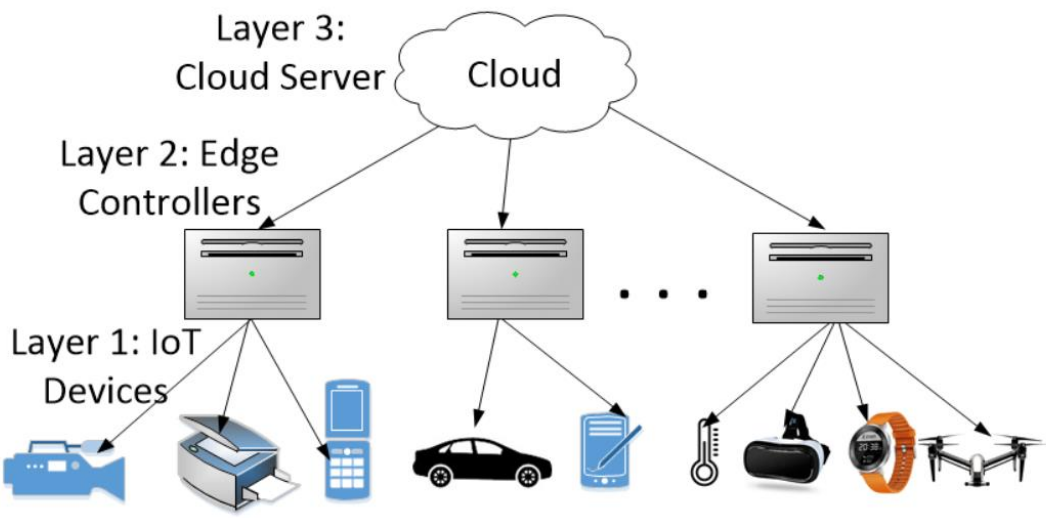

Figure 1: Edge Computing Hierarchy

WORKING OF EDGE COMPUTING IN IOT

The working of edge computing in IoT systems follows a smart data handling approach where data is processed as close as possible to the source. Instead of blindly transmitting all raw sensor data to cloud, the system performs intelligent operations at the edge layer.This reduces delay, saves bandwidth and allows faster decision making.

The complete workflow can be understood in the following steps.

1. Sensors Collect Raw Data

IoT devices such as sensors, cameras, wearable gadgets, smart meters and industrial machines continuously sense the environment and generate raw data.

Examples of raw data:

- Temperature, humidity, pressure readings
- Video streams from surveillance cameras
- Heart rate and blood pressure from wearable devices

- Vibration data from industrial motors
- GPS location data from vehicles

This data is usually unstructured and very large in volume. Direct transmission of this raw data to cloud is inefficient and costly.

2. Edge Node Performs Filtering, Aggregation and Analysis

The raw data from sensors is first sent to nearby edge node such as IoT gateway or local server. This edge node has enough computational capability to perform initial data processing.

Operations performed at edge:

- **Filtering:** Removing unnecessary or duplicate data
- **Aggregation:** Combining multiple sensor readings into summarized form
- **Pre-processing:** Noise removal, normalization, formatting
- **Local analytics:** Running lightweight machine learning models or rule-based logic

For example, instead of sending every second temperature reading, edge device may send only hourly average value or only abnormal readings.

3. Important Data Forwarded to Cloud

After processing, only meaningful, summarized or abnormal data is transmitted to cloud for further storage and deep analysis.

This step reduces:

- Network bandwidth consumption
- Cloud storage requirement
- Data transmission cost
- Cloud is used mainly for:
- Historical data storage
- Training complex AI/ML models
- System-wide analytics and reporting

4. Real-Time Decisions Taken Locally

One of the main advantages of edge computing is the ability to make instant decisions without waiting for cloud response.

Edge nodes can trigger actions such as:

- Sending alert to user
- Activating actuator or alarm
- Adjusting machine parameters
- Controlling traffic signals

This is very useful in time-critical IoT applications.

Example: Smart Traffic Management System

In a smart traffic system:

1. CCTV cameras capture continuous video of roads
2. Video feed is sent to edge server installed at traffic junction
3. Edge server uses image processing to detect vehicle density and congestion
4. Based on analysis, traffic signal timing is adjusted immediately
5. Only traffic statistics or alerts are sent to cloud for record keeping
6. Here, full video is not sent to cloud which saves bandwidth and enables quick response.

5. Enabling Technologies

- 5G and high-speed networks
- Artificial Intelligence at edge
- Containerization (Docker, Kubernetes)
- Software Defined Networking (SDN)
- Network Function Virtualization (NFV)

These technologies make edge nodes more intelligent and flexible.

APPLICATIONS OF EDGE COMPUTING IN IOT

1. Smart Cities

Traffic control, street lighting, waste management using edge nodes.

2. Healthcare

Wearable sensors monitor patient vitals and edge device analyze data instantly.

3. Agriculture

Soil sensors and weather data processed locally for irrigation decisions.

4. Industrial IoT

Machines monitored in factories, faults detected at edge.

5. Autonomous Vehicles

Real-time decision-making is done in vehicle edge system.

Table 2: Edge Applications

Domain	Role of Edge Computing
Healthcare	Instant health monitoring
Smart City	Traffic and surveillance control
Agriculture	Smart irrigation decisions
Industry	Predictive maintenance
Transport	Autonomous driving decisions

ADVANTAGES OF EDGE COMPUTING

- Reduced latency
- Improved privacy
- Lower bandwidth usage
- Real-time processing
- Better reliability
- Energy efficient

CHALLENGES IN EDGE COMPUTING

- Limited storage and processing power at edge
- Security threats at distributed nodes
- Management of large number of edge devices
- Interoperability issues
- Scalability problems

SECURITY AND PRIVACY ISSUES

Since edge devices are distributed, they are more vulnerable to attacks. Proper encryption, authentication and access control must be applied.

CASE STUDY EXAMPLE

In a smart factory, sensors generate vibration data of machines. Edge gateway analyzes data and detects abnormal pattern. Only summary report is sent to cloud. This reduces delay and network load.

FUTURE RESEARCH DIRECTIONS

- Integration of AI and edge
- Edge orchestration using blockchain
- Energy efficient edge devices
- Standard protocols for interoperability
- Edge-cloud collaboration models

COMPARISON OF EDGE MODELS

Table: 3

Model	Location of Processing	Example Use
Cloud	Remote data centers	Big data analytics
Fog	Near network gateways	Smart grid
Edge	Near devices	Healthcare monitoring
Mist	Sensor level	Wearable devices

DISCUSSION

Edge computing is not replacing cloud but complementing it. Hybrid model is best for IoT. Real-time tasks are done at edge while heavy tasks at cloud. Many industries are adopting this model rapidly.

CONCLUSION

Edge computing is becoming very important technology for IoT systems. It helps to reduce latency, improve privacy and enable real-time applications. Many domains such as healthcare, agriculture, industry and smart cities are benefitting from edge architecture. Although there are challenges like security and management, future research will solve these problems. Combination of edge and cloud will define future IoT ecosystem.

REFERENCES

1. Shi, W., et al., "Edge Computing: Vision and Challenges," IEEE Internet of Things Journal, 2016.
2. Satyanarayanan, M., "The Emergence of Edge Computing," Computer, IEEE, 2017.
3. Yi, S., Li, C., Li, Q., "A Survey of Fog Computing," IEEE, 2015.
4. Bonomi, F., et al., "Fog Computing and Its Role in IoT," MCC Workshop, 2012.
5. Chiang, M., Zhang, T., "Fog and IoT: An Overview," IEEE IoT Journal, 2016.
6. Roman, R., Lopez, J., "Security Issues in Edge and Fog Computing," Future Generation Computer Systems, 2018.
7. Zhang, W., et al., "Edge Intelligence for IoT," IEEE Network, 2019.
8. Abbas, N., et al., "Mobile Edge Computing: A Survey," IEEE Internet Computing, 2018.
9. Mach, P., Becvar, Z., "Mobile Edge Computing: Survey and Research Outlook," IEEE Communications Surveys, 2017.
10. Deng, R., et al., "Optimal Workload Allocation in Fog-Cloud Computing," IEEE Transactions, 2016.

Cite as:

Sabita Sharma, Devansh Kulkarni, Megha Patil, Sambhu Rawat (2026). Edge Computing for IoT Systems: Architecture, Applications, Challenges and Future Directions. Journal of Internet of Things and Information Technology Things. 9(1), 52-63.

https//doi.org/10.5281/zenodo.19275911

