

Article

Future of Edge Computing for Real Time Devices in Internet of Things

Mir Hassan ^{1,2*}, Jincai Chen ^{1†}, Tariq Khan ³, Umer Zukaib ⁴, and Shoaib Ali ⁵

¹ Wuhan National Laboratory for Optoelectronics, Huazhong University of Science and Technology, Wuhan, China; hassanmir@hust.edu.cn, jcchen@hust.edu.cn

² Institute of Data Science and Digital Technologies, Vilnius University, Vilnius, Lithuania

³ Department of Information Engineering, University of Politecnico Delle Marche, Ancona, Italy

⁴ Department of Computer Science, COMSATS University, Abbottabad Campus, Pakistan

⁵ Department of Computer Science, Virtual University, Pakistan

* Correspondence: hassanmir@hust.edu.cn; jcchen@hust.edu.cn

† These authors contributed equally to this work.

Abstract: Rapid growth of IoT applications and their interference in our daily lives led to many different IoT devices which generates enormous data. The IoT devices' resources are very limited, so storing and processing IoT data in the devices is very inefficient. Several resources of cloud-computing are efficiently used to handle some IoT resources issues. While using resources in the cloud centers cause some other issues, like latency in the IoT applications, which are time-critical. Thus, the technology of edge cloud has evolved recently. This technology permits storage and data processing at the network edge. This paper studies edge computing in-depth for timeless sensitive devices in IoT. In-depth, cutting-edge IoT computing systems (ECAs-IoT) are evaluated and characterized in this paper according to numerous criteria, such as information placement, improvisation facilities, reliability, and data visualization. Moreover, according to distinctive properties, the paper aims at comparing each structure in detail. The paper also highlights the significant limitations of the new ECAs-IoT and recommends solutions to them. The studies also introduce and propose solutions to some of the most important restrictions of the current ECAs-IoT. Consequently, in the edge computing domain, this survey outlines the IoT implementations. Lastly, with the use of IoT implementations for ECAs-IoT, the paper suggests four distinct scenarios.

Keywords: Internet of Things, Sensors, Real-Time, Edge Computing

2. Introduction

With the universal adoption of detectors in the modern world, through information. Sharing sensors individually are gradually connected to the Internet of Things (IoT). IoT technology is now widely used in several sectors, such as smart cities, smart houses, open surgical devices, or environmental attitudes [1]. Vast sums of items are linked to the internet, and by 2020 the organization plans to operate 50 billion IoT computers [2]. IoT data is continuously transferred from applications to the primary storage facility, typically situated in the cloud center. The immense data processing overhead cannot be overlooked, as the cloud reduces the responsibility of sensors and processors for analysis. The total number of IoT-connected operating systems exceeded 11.2 billion in 2018 and is expected to rise to 20 billion by 2020, bringing data growth.

Edge computing (EC) is essential to meeting these requirements by integrating cloud-like technology at the end devices. The EC does not constitute a fast remedy. On the one hand, IoT systems under the EC have been significantly improved in many areas (computing offloading, accurate placement, real-time computation) to gather intelligence to reduced data acquisition close to end-users. On the other side, EC introduces new safety obstacles and expands the surfaces of three-dimensional threats [3]. ENs are mounted on the network's edge in different sites, making centralized control systems challenging to unify. Due to the physical nature, the computational functionality of ENs is limited, unlike cloud storage, which ensures that heavy-duty security controls are not suitable for ENs and



Citation: Lastname, F.; Lastname, F.; Lastname, F. Title. *Preprints* 2022, 1, 0. <https://doi.org/>

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

centrally controlled attacks, for example. EC uses various innovations, including cellular sensor networks, storing cell data, cloud computing, and processing smartphone data. In this heterogeneous setting, it is hard to formulate a single security architecture and ensure continuity between different security realms in security policy.

Most modern protection systems are focused on methods and algorithms that follow a single pattern. New security alternatives are created by the rise of artificial intelligence [4]. The Intrusion Detection System (IDS) detects threats by hacked ENs by finding anomalous network conduct. Machine learning (ML) extracts disruptive access patterns from previous data set training. Distributed machine learning (DML) ensures the ENs have to transfer the requirements to other cooperative learning ENs after each preparation. It decreases the risk of data loss and the strain on the network during transmission. Only nodes and information under their authority can access each authorized node. ENs must be categorized into various groups according to approval. The algorithm classifies network-connected ENs on low-IoT-rights computers [5]. Although several studies have been performed on AI and EC mergers, there is still little discussion and investigation of EC-based IoT protection in AI. The ECA-IoT concept is presented in this paper, and the current ECA-IoT and future research prospects are examined. ECA-IoT is defined as a computer architecture that includes computers, applications, network protocols, and infrastructure related to providing some services for IoT, edge, and cloud computing. We have graded the ECAs-IoT as per criteria. One criterion is challenge-based. ECA-IoT discusses specific tasks, such as optimal scheduling management, allocation of tasks and procedures, operation orchestration, and large-data analysis. Other criteria are technology-oriented. Like software-defined networking (SDN) and machine learning, ECA-IoT was classified depending on the technologies used [6].

Besides offering a description of edge computing software with a particular emphasis on IoT edge computing, a new core word (ECA-IoT) has been developed and produced (edge, cloud, and IoT). Taxonomy is given for IoT applications based on various criteria, including the framework feature, structure of the IoT application, amount of traffic, latency susceptibility, hence the need for cloud data acquisition at the network edge [7]. For IoT implementations using the ECAs-IoT, four new possibilities are proposed. Use, Change, Blend, and Fresh is known as the suggested scenarios. Five core libraries were used for the initial search: IEEE Explore, ACM Digital Archive, Elsevier, Springer, and Google Scholar. IEEE Explore, the crucial database for this study, contained much of the related documents. Responding to RQ1 was the initial browsing of related posts; then, we found that ECAs-IoT had addressed many IoT problems [8].

3. Methodology

Edge computing includes placing micro statistics facilities or maybe small, purpose-constructed high-overall performance statistics analytics machines in faraway places of work and places to advantage real-time insights from the statistics collected or to sell statistics thinning at the threshold, with the aid of using dramatically decreasing the number of statistics that desires to be transmitted to a crucial statistics center. Without transporting needless statistics to a crucial statistics center, analytics at the threshold can simplify and substantially velocity evaluation at the same time as additionally slicing costs. "Edge computing pushes packages, statistics, and computing power (services) far from centralized factors to the logical extremes. Edge computing replicates fragments of records throughout dispensed networks of net servers, which can be vast. As a topological paradigm, side computing is likewise known as mesh computing, peer-to-peer computing, autonomic (self-healing) computing, grid computing, and the usage of different names implying non-centralized, node less availability." "To make the certain desirable overall performance of broadly dispersed dispensed services, big corporations usually put into effect side computing with the aid of using deploying Web server farms with clustering. Previously to be had handiest to a very big company and authorities' corporations, generation development and price discount for big-scale implementations have made the generation to be had to small

and medium-sized businesses. The goal end-consumer is any Internet customer using business Internet software services. Edge computing imposes sure barriers at the selections of generation platforms, packages, or services, all of which want to be mainly evolved or configured for side computing.”

4. Edge Computing

We will clarify edge computing, Implementing edge-computing, and edge intelligence in this section.

4.1. Edge Computing Concepts

Edge computing is a modern distributed IT design that transfers data storage, services, and computing applications largely or entirely from centralized nodes to close end-users. IoT systems produce significant amounts of data; the data produced could exceed 500 zettabytes [9]. It adds to difficulties associated with bandwidth, space, and transmission. Edge-Computing technology aims to reduce lag and reaction time [10] in real-time applications. Edge computing applies cloud computing technology to the network’s edge. The edge and the cloud are complementary; edge computing facilitates the network’s provision, and the cloud handles the network. The edge unit conveys storage, power, and contact near the end-user. The network is also strengthened in multiple ways by using edge processing on the network. Apprehension is the edge system mindful of the specifications of consumers [11].

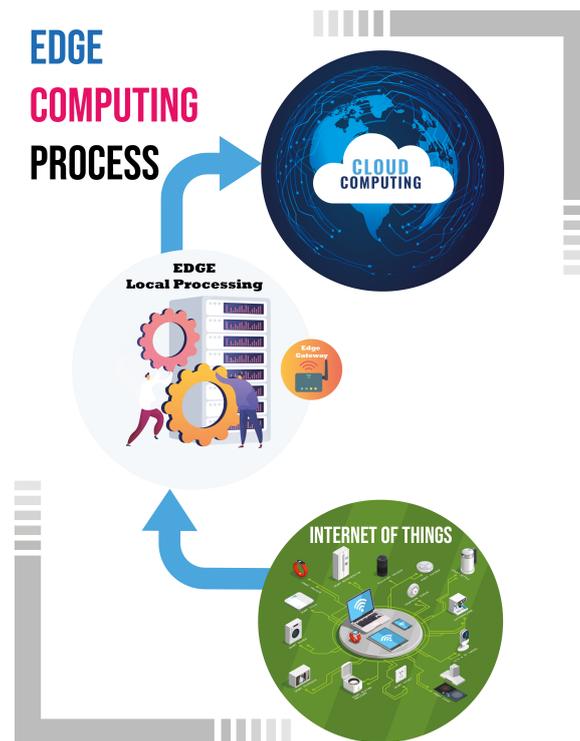


Figure 1. Working of Edge Computing.

For example, patients’ physical health is monitored in the e-health system via IoT devices, especially in emergency patient circumstances. The distribution of computer resources is adjusted based on the user’s health risk score [12]. Dexterity is also used because data processing and storage are carried out next to the end-user [13]; experimenting with edge devices and clients is faster and easier. Latency edge computing helps time-critical technologies by making it easy for near-end users to capture data and process data, allowing IoT applications to make better and smarter choices [14] and competence is the edge system takes full advantage of various assets by assigning memory, computation, and

managing tasks to the existing resources at every place between the end-user and the cloud [15]. It helps to allow distributed edge computing resources for IoT applications to be used optimally.

4.2. Implementation of Edge-Computing

There are three implementation modes for Edge computing: cloud, mobile edge computing (MEC), and fog computing [10]. They differ in terms of the node's characteristics, architecture, and venue. The differences between these implementations are indicated in this subsection. "A high-level description of the overall DRUID-NET system is shown in figure 1. The architecture follows the model of NFV/SDN and divides the information flow into planes of control and data. The IoT applications are deployed at the lowest tier, and the generated workload (data flow) can be offloaded at the upper level of Edge Computing for further processing. In this layer, as a virtualized service, every part of the application is supported. A virtualized service refers to basic IoT functionalities, such as route planning and image recognition, or control components such as learning algorithms or solvers for optimization, as seen in the figure. In order to construct workload-resource profiles, update the output model for each application, and understand the input management process for the allocation of resources, the modelling and control system collects information (power flow) about the state of the computing and network networks at the edge computing level, while at the same time applying a resource-aware control strategy f (control flow). This holistic approach facilitates complex simulation of the programme, taking into account different contextual details. In addition, in virtualized services, the controller co-design considers the resource allocation algorithms as device components."

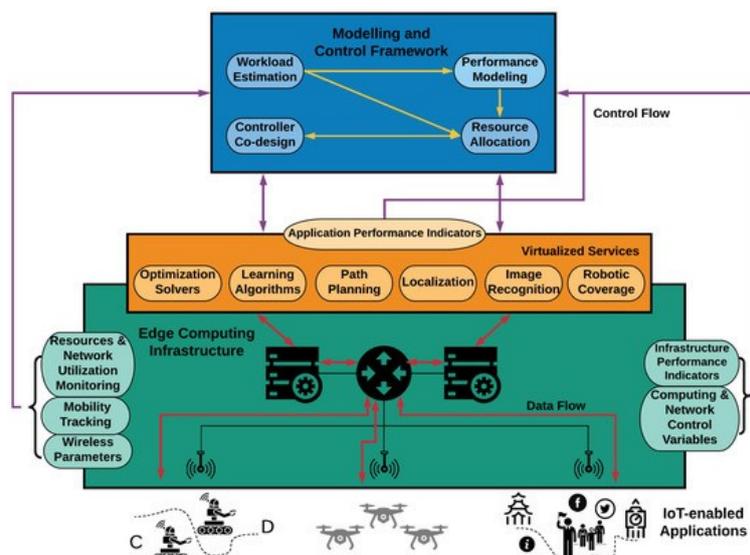


Figure 2. Conceptual Architecture of Edge Computing.

4.3. Edge Intelligence

Standard edge computing systems have limited knowledge capacities. These tools are responsible for local data analysis, such as the extraction [16] feature and data conversion to cloud servers [17]. The user's anonymity, security, and performance are maintained by data collection near the end-user. During the transmission process, the data is not exposed to noise. Data collection at the edge restricts Internet bandwidth consumption.

5. Large-Scale implementations of IoT

IoT installations can be small [18,19] or huge in number [13,18,19]. Heterogeneity and broad IoT outcomes usually describe IoT-LSDs. The size of the data they obtain is improved by increasing the number of connected IoT devices. It was estimated by Shi

[20] that the data generated could surpass 500 zettabytes. Also, it is estimated that transit traffic will reach 10.9 zettabytes. In specific, these features can lead to issues that cannot be solved immediately by the use of conventional approaches [19]. Cloud-assisted strategies, especially for time-critical applications, are often inadequate to overcome these challenges. Moving IoT data to cloud-based data centers from IoT machines increases bandwidth and overhead connectivity.

Characteristics	Conventional cloud computing	Edge cloud and edge computing
Major applications	Most of the current mainstream cloud-involved applications	Applications on IoT, VR, AR, smart homes, smart cities, smart energy, smart vehicles, etc.
Availability	A small number of large-sized datacenters	A large number of small-sized datacenters
Proximity of services and resources; Data processing location	Usually in remote datacenters and far from users	At the edge close to the users
End-to-end latency	High, due to the distance between the edge and remote datacenters	Low, due to proximity to the users
Backbone network bandwidth consumption	High, since huge data need to be transferred to the datacenters first	Low, since data are locally processed and stored in edge cloud
Scalability	Scalable at center	Scalable both center and edge
Security (e.g., attacks on data enroute)	Data subject to attack due to long-distance transmission; Physical security depends on large facilities	Lower risk for enroute attacks; Physical security varies and different mechanisms needed

Figure 3. Cloud and Edge Computing.

6. Mapping of ECAs-IoT to IoT 5/3-Layer Models

We map the ECAs-IoT to two existing IoT layered models in this chapter: 5- and 3-layer models. Mapping involves separating each ECA-IoT into its components and, where appropriate, matching each component to the corresponding layer of each IoT model. Mapping helps the features, and gaps of each ECA-IoT be identified to support IoT layer models.”

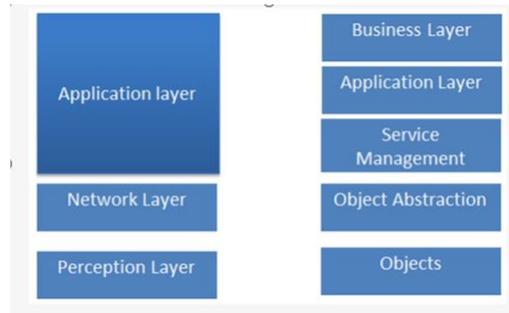


Figure 5. 3 layer and 5-layer model for IoT architecture

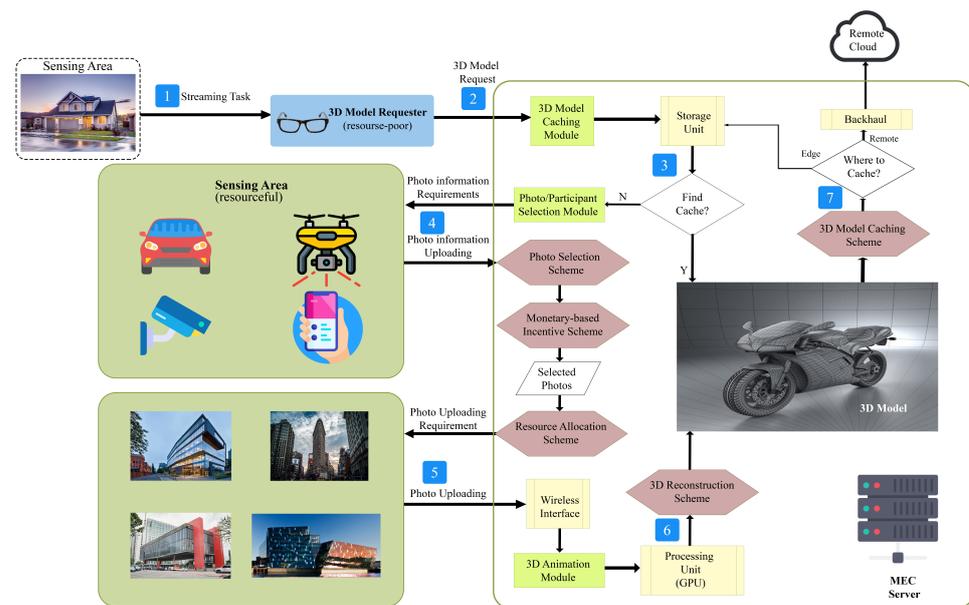


Figure 4. Cloud and Edge Computing.

6.1. Existing IoT Layer Templates

Many IoT reference frameworks have been proposed, like IoT-A, originating from industry requirements. In Figure 4, the two most common structures are shown. The following layers in the three-layer architectural design: a view layer, a network layer, a system layer, and a data processing layer. The importance of providing a prototype system is to specify the kind of data that helps analysts to identify new procedures in each layer.

IoT resources responsible for producing IoT data, such as sensors and smartphones, form the entity layer. A layer of object abstraction passes IoT data to the service management layer from the object layer. Service-management layer that helps programmers to link resources to interact with heterogeneous products with their requestors. Application layer, which provides consumers high-level features.

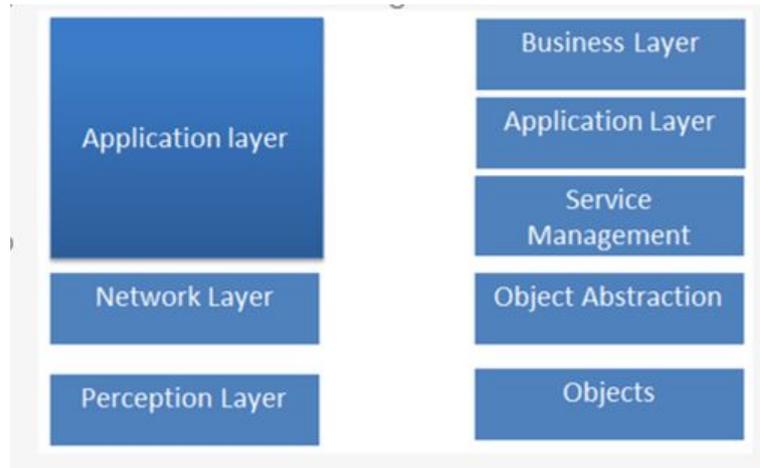


Figure 6. Cloud and Edge Computing.

6.2. Mapping IoT Layer Models to ECAs-IoTs

The mapping of IoT layers with architectural components helps researchers recognize the components involved in any alteration or modification they choose to construct [21]. For example, at the bottom of two layers of the five-layer model IoT, the TTM architecture operates. First, the section presents the two IoT layered models. It includes a comprehensive overview of the ECAs-IoT for these two models. The next section provides a more detailed review.

6.3. Layer-Mapping Analysis

Mapping helps the fast identification of the IoT capabilities of each ECA. Some variants concentrate on bottom layer features only, including TTM and SDNB ECAs. For some ECAs, the service layers responsible for dealing with heterogeneous artifacts lack assistance. In short, visualization has several advantages: It enables researchers to identify each ECA-capabilities IoT's and functionality to support layered IoT models. It helps detect holes inside each ECA-IoT to assist layered IoT models. For example, suppose an ECA-IoT does not cover an IoT model layer. In that case, this means that additional elements, such as using an external protocol within the ECA-IoT or expecting the IoT implementation feature, need to safeguard the functionality. Existing IoT layered models do not apply to the edge computing concept. This section ties IoT layered models with edge computing. This component integrates IoT layered models of edge computing.

7. Results and Discussions

IoT literature analyses and interventions show the overwhelming role of technology in IoT programs, often driven by technological intervention in the business model rather than creativity. There are some essential concerns, notably in protection and privacy, about the risks of IoT growth, so industry and government have begun to address these concerns. What makes IoT interesting at the end of the day is that we do not yet know the exact use of cases that would significantly impact our lives.

7.1. Fog vs Cloud computing

However, with the integration of separate vertical IoT domains to offer more precise options for users, it is no longer sufficient to provide an independent context. Although researchers have investigated many context-sharing approaches, there is no effort to identify the Edge-centric Context Sharing Architecture for IoT environments that mitigate context-sharing requirements and provide context-conscious protection features. In this way, this research defines an architecture of edge-oriented, context-sharing systems based on an edge-to-fog strategy to decrease overhead network and latency.

Table 1. Comparison of edge base applications

	Fog Computing	Edge Computing	Cloudlet Computing
Devices used as a node	Routers and gateways	Servers that operate at base stations	In a box data centre
Location of nodes	Varying between Server and End Devices	Controller/Macro Base Station Radio Network	Installing Local /Outdoor
structure	Based on Fog Abstraction Layer	Based on Handheld Orchestrator	Based on Cloudlet Agent
Awareness	Medium Scale	High	Low
Mechanism of data accessing	Hops One or Many	Two hops	single
proximation	Wi-Fi, Bluetooth, Broadband Networks	Partial	Partial

8. Conclusion

As a new computational paradigm that provides various solutions to traditional cloud issues, EC would promote the IoT field's development and expand the IoT technology ecosystem's diversity. Current IoT characterizations tend to a broken picture and a lack of common awareness of IoT architectures and their elements. We have seventeen features identified. As a new computational framework that provides various solutions to traditional cloud issues, EC would promote the IoT field's development and expand the IoT technology environment's diversity. AI's strong learning ability helps the device to more effectively and reliably detect malicious threats. According to the IoT problems they intend to address, this survey classified the ECAs-IoT. It involves architectures focused on data location that strive to control. IoT implementations that are based on the role that the design supports are classified in the document. The study revealed that ECs and new ECAs should be seen in four specific cases.

8.1. Future Work of Edge Computing

5G is one of the drivers of edge computing because it allows for an increased number of factual assets or processing variables that can be interconnected, suggesting an exponential increase in the number of facts to be processed. An unnecessary amount of current "web sites to cloud" links and requests for facts that process tons to the source will end quickly. 5G also allows loads of decrease latencies, a key issue for a few new packages, and some other component to be had towards where it's far eating up or produced in processing energy. "There will be an open future for edge computing. Via artificial intelligence and machine learning, Edge will converge with data to provide insights into behavior that will help companies and their clients. It is regarded, after all, like any other position where applications can be seamlessly placed with consistency and without compromise. I expect an increased emphasis on defense. Most IoT apps cover the edge and the cloud and, not least because of the scale of the possible attack surface, have a huge effect on the security of both. There can be nothing approved. New and innovative threats are constantly being published, making robust edge protection at its heart a must with minimal power. To accomplish this, an emphasis on zero trust is necessary."

8.2. Reliability of Edge Computing

"Edge computing decreases data flow to and from the primary network by processing data closer to the source and prioritizing traffic, resulting in lower latency and faster overall speed. For efficiency, physical distance is also important. Organizations can significantly reduce the distance data needs to move before services can be delivered by centers geographically closer to end users by better distributing processing. For their clients who expect access to their content and on-demand applications anytime, anywhere, these edge networks ensure a quicker, smoother user experience. With so many network-

connected edge computing devices and edge data centers, for someone who does not close the service completely, it becomes even more difficult. Therefore, the products and information needed to successfully integrate IoT edge computing devices and edge data centers are robust edge architectures that can provide unparalleled reliability.”

9. Funding

This work was supported by the National Key R&D Program of China (No. 2018YFC1604000) and the National Natural Science Foundation of China (No. 61572374 & No. U163620068).

10. Conflict of Interest

The authors have no conflict of interest with any organization.

1. Botta, A.; Donato, W.; Persico, V.; Pescapé, A. Integration of Cloud computing and Internet of Things: A survey. *Future Gener. Comput. Syst.* **2016**, *56*, 684–700.
2. Babu, S.M.; Lakshmi, A.J.; Rao, B.T. A study on cloud based Internet of Things: CloudIoT. 2015 Global Conference on Communication Technologies (GCCT), 2015, pp. 60–65. doi:10.1109/GCCT.2015.7342624.
3. Yang, Z.; Zhou, Q.; Lei, L.; Zheng, K.; Xiang, W. An IoT-cloud Based Wearable ECG Monitoring System for Smart Healthcare. *Journal of medical systems* **2016**, *40*, 286. doi:10.1007/s10916-016-0644-9.
4. Hossain, M.S.; Muhammad, G. Cloud-assisted Industrial Internet of Things (IIoT) – Enabled framework for health monitoring. *Computer Networks* **2016**, *101*, 192–202. Industrial Technologies and Applications for the Internet of Things, doi:https://doi.org/10.1016/j.comnet.2016.01.009.
5. Roman, R.; Lopez, J.; Mambo, M. Mobile edge computing, Fog et al.: A survey and analysis of security threats and challenges. *Future Generation Computer Systems* **2018**, *78*, 680–698. doi:https://doi.org/10.1016/j.future.2016.11.009.
6. Lee, I.; Lee, K. The Internet of Things (IoT): Applications, investments, and challenges for enterprises. *Business Horizons* **2015**, *58*, 431–440. doi:https://doi.org/10.1016/j.bushor.2015.03.008.
7. Aburukba, R.O.; AliKarrar, M.; Landolsi, T.; El-Fakih, K. Scheduling Internet of Things requests to minimize latency in hybrid Fog–Cloud computing. *Future Generation Computer Systems* **2020**, *111*, 539–551. doi:https://doi.org/10.1016/j.future.2019.09.039.
8. Zhou, J.; Cao, Z.; Dong, X.; Vasilakos, A.V. Security and Privacy for Cloud-Based IoT: Challenges. *IEEE Communications Magazine* **2017**, *55*, 26–33. doi:10.1109/MCOM.2017.1600363CM.
9. Mao, Y.; You, C.; Zhang, J.; Huang, K.; Letaief, K.B. A Survey on Mobile Edge Computing: The Communication Perspective, 2017, [arXiv:cs.IT/1701.01090].
10. Chen, M.; Li, W.; Hao, Y.; Qian, Y.; Humar, I. Edge cognitive computing based smart healthcare system. *Future Generation Computer Systems* **2018**, *86*, 403–411. doi:https://doi.org/10.1016/j.future.2018.09.009.
11. Yi, S.; Qin, Z.; Li, Q. Security and Privacy Issues of Fog Computing: A Survey. 2015, pp. 685–695. doi:10.1007/978-3-319-21837-3_67.
12. Shi, W.; Dustdar, S. The Promise of Edge Computing. *Computer* **2016**, *49*, 78–81. doi:10.1109/MC.2016.145.
13. Shi, W.; Cao, J.; Zhang, Q.; Li, Y.; Xu, L. Edge Computing: Vision and Challenges. *IEEE Internet of Things Journal* **2016**, *3*, 637–646. doi:10.1109/JIOT.2016.2579198.
14. Dolui, K.; Datta, S.K. Comparison of edge computing implementations: Fog computing, cloudlet and mobile edge computing. 2017 Global Internet of Things Summit (GIoTS), 2017, pp. 1–6. doi:10.1109/GIOTS.2017.8016213.
15. Li, Y.; Wang, W. Can mobile cloudlets support mobile applications? IEEE INFOCOM 2014 - IEEE Conference on Computer Communications, 2014, pp. 1060–1068. doi:10.1109/INFOCOM.2014.6848036.
16. Anjana, A. On improving fault tolerance of IoT networks through Butterfly Networks implemented at Services Layer. *International Journal of Advanced Trends in Computer Science and Engineering* **2020**, *9*, 2096–2115. doi:10.30534/ijatcse/2020/184922020.
17. Kitagami, S.; Ogino, T.; Suganuma, T.; Shiratori, N. Proposal of a Multi-agent Based Flexible IoT Edge Computing Architecture Harmonizing Its Control with Cloud Computing. 2017 Fifth International Symposium on Computing and Networking (CANDAR), 2017, pp. 223–229. doi:10.1109/CANDAR.2017.28.

-
18. Fernandez, D.M.; Kleingartner, J.; Oliphant, A.; Bowman, M.; Torregrosa, A.; Weiss-Penzias, P.S.; Zhang, B.J.; Sorensen, D.; Cohen, R.E.; McKinley, G.H. Fog Water Collection Effectiveness: Mesh Intercomparisons. *Aerosol and Air Quality Research* **2018**, *18*, 270–283. doi:10.4209/aaqr.2017.01.0040.
 19. Vögler, M.; Schleicher, J.; Inzinger, C.; Nastic, S.; Sehic, S.; Dustdar, S. LEONORE – Large-Scale Provisioning of Resource-Constrained IoT Deployments. 2015 IEEE Symposium on Service-Oriented System Engineering, 2015, pp. 78–87. doi:10.1109/SOSE.2015.23.
 20. Mainetti, L.; Patrono, L.; Stefanizzi, M.L.; Vergallo, R. A Smart Parking System based on IoT protocols and emerging enabling technologies. 2015 IEEE 2nd World Forum on Internet of Things (WF-IoT), 2015, pp. 764–769. doi:10.1109/WF-IoT.2015.7389150.
 21. Naas, M.I.; Parvedy, P.R.; Boukhobza, J.; Lemarchand, L. iFogStor: An IoT Data Placement Strategy for Fog Infrastructure. 2017 IEEE 1st International Conference on Fog and Edge Computing (ICFEC), 2017, pp. 97–104. doi:10.1109/ICFEC.2017.15.