2020 3rd International Conference on Computer and Informatics Engineering (IC2IE)

# The Development of IoT-Smart Basket : Performance Comparison between Edge Computing and Cloud Computing System

Nandiwardhana Waranugraha, Muhammad Suryanegara
*Department of Electrical Engineering*
*Universitas Indonesia*
Depok, Indonesia
nandiwardhana.waranugraha@ui.ac.id, m.suryanegara@ui.ac.id

***Abstract*—This paper aims to develop the Internet-of Things (IoT) Smart-Basket, working on 2 different systems, i.e. Edge Computing and Cloud Computing. To identify the best system, we compare the performance between "Edge Computing" system and "Cloud Computing" system. The system consists of Raspberry Pi hardware and webcam. Python, TFLite, OpenCV, and Google Cloud Vision API software to detect shopping objects. The object detection results are calculated and sent to end-users through the Telegram application. Discussions are presented concerning the Time Performance and RSSI Value between two systems. The results show "Edge Computing" systems have a more stable system with an average processing time of 1.74 sec on Line-of-Sight (LOS) condition and 1.75 sec on Non-Line-of-Sight (NLOS) condition compared to "Cloud Computing" systems with an average processing time of 10.46 sec on LOS condition and 5.36 sec on NLOS condition.**

***Keywords— internet of things (IoT), edge computing, cloud computing, smart-basket***

## I. INTRODUCTION

Nowadays, every supermarket still uses a shopping cart or shopping basket as a tool to collect items. Customers must put the items they want to buy and then proceed to the cashier to checkout. After that, the customer must wait to get a bill or receipt, which sometimes takes a long time [1].

Based on the Time Use Institute, the average shopping time takes about 41 minutes. If multiplied by an average of 1.5 trips per week, more than 53 hours per year are spent shopping at supermarkets. Therefore, Smart-Basket based on IoT solutions is proposed because they are quite effective for self-service, which can shorten the time [2].

In the last decade, the case of computing, control, and data storage of IoT devices using the Cloud Computing system has become a trend. However, now the Cloud Computing system is facing many large challenges in meeting new specific requirements for more widely IoT scenarios [3]. On the other side, the Edge Computing system is starting to be more used than the Cloud Computing system. The Edge Computing system allowed data processing locally and distributed. The implementation of Edge Computing is widely applied in several sectors. Such as smart cities, smart home, smart manufacture, and smart groceries.

This paper develop the IoT-Smart-Basket working on 2 (two systems), i.e. "Edge Computing" system and "Cloud Computing" system, then comparing their performance. Both systems utilize the Wi-Fi for Internet of Things (IoT) as the main communications to the end-user. The system consists of Raspberry Pi hardware and webcam. Python, TFLite, OpenCV, and Google Cloud Vision API software to detect shopping objects. We conduct testing by measuring the parameters of time performance and RSSI Value. It is expected the results will help us to understand which system showing the ideal performance for the Smart-Basket.

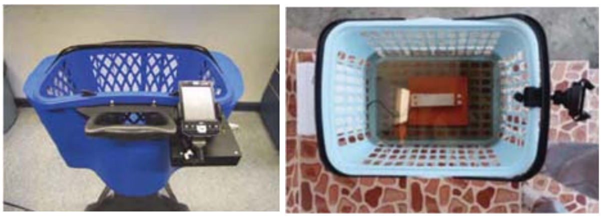
This paper comprises five sections. Section I is the introduction; Section II presents the underlying theories and relevance to other research works. Section III provides the proposed system design, while Section IV analyzing the result of the test scenarios. Section V concludes the paper.

## II. UNDERLYING THEORIES AND RELATED WORKS

### *A. Related Works of Smart-Basket*

Research on Smart-Basket has been carried out in recent years. For the example [4], [5] is Smart-Basket using RFID and ZigBee technology as shown in Fig. 1. The technology enables the creation of centralized and automated networks. From the study of this technology, the shopping basket has three components. The first component is the shopping basket, the second component is the communication system, and the third is a centralized system. In [4], [5], the system used the Cloud Computing system aggregating information from several smart-basket devices. Smart-Basket will have an installed system such as an RFID reader and all items in supermarkets that already have a barcode. The barcode will be scanned using an RFID reader when the buyer saves his purchases on the Smart-Basket. Shopping results in the form of receipts that will be stored in the device memory. Then the shopping receipt results will be displayed on the LCD screen of the device, and the customer can pay for the purchase with several methods offered.

In other studies, Smart-Basket also uses camera sensors as a barcode detection tool [6-7]. The camera will take a barcode image from the shopping object, and the device will process the barcode. By processing the barcode, the device will find out what items are purchased and can calculate the customer's shopping results. The use of Artificial Intelligence has also begun to be used as a mechanism to detect shopping objects. Research at Stanford [8] shows an algorithm that can recognize shopping objects by analyzing objects photographed with Artificial Intelligence offerings embedded in the Smart-Basket system.



(a) (b)

Fig 1. Examples of Smart-Basket using RFID Technology (a) and using a Camera (b) [5] [7]

*B. IoT, Cloud Computing and Edge Computing*

Internet of Things or commonly abbreviated as IoT is a network of physical objects that are embedded with sensors, software, and other technologies that aim to connect and exchange data with other system devices via the internet [9]. IoT is not only about devices connected to the internet [10-11], but also about hardware, software, connectivity, communication protocols, middleware and technological processes (big data, data analytics, cloud computing, edge computing, fog computing, IoT platforms, IoT gateways, etc.) needed to do something with the Internet of Things [12].

In general, IoT devices use Cloud-centric Internet of Things (CIoT) architecture systems or commonly called Cloud Computing systems where physical objects are represented in the form of Web services managed by servers on the Internet on one core network [13]. Cloud-Centric architecture is above the IoT device and below the application layer. Cloud Computing has the advantage that users will be more flexible and have high scalability. Cloud Computing offers core infrastructure, platforms, API software and storage media. Developers can also develop software tools in the Cloud such as data mining, machine learning and data visualization.

The Edge Computing architecture represents an approach between the physical layer and the transport layer. Monitoring and preprocessing occur at the edge of the network before data is transmitted to the cloud [14]. Edge Computing is a new paradigm where resources on the server are placed at the end of the network closer to the device, sensors and end-users. Terms such as "cloudlets", "microdata center", and "fog" are often used in the literature to refer to the type of small hardware located at the edge of the network. This paradigm breaks that there must be a large data center in Cloud Computing[15].

## III. System Design

This paper develop the IoT-Smart-Basket working on 2 (two) different systems, i.e. “Edge Computing” system and “Cloud Computing” system. The device does not require RFID or barcode scan to recognize items or object but uses AI that has been trained to detect certain object, especially products in supermarkets. The concept is that, shopping object detection will be processed using such two different systems. The systems are:

- “System A”: The Smart-Basket working on the basis of “Edge computing”. The main feature of this system is that the object detection processing will be operated in the shopping basket device.
- “System B”: The Smart-Basket working on the basis of “Cloud Computing”. The main feature of this system is that the object detection processing will be operated in the cloud server. Image of the shopping object will be sent from smart-basket to the cloud-server.

Fig. 2 shows the proposed design of the Smart-Basket using two different systems. The proposed systems use Raspberry Pi 4 as a Smart-Basket device and webcam as a visual sensor. End-users will send commands to Smart-Basket to detect objects using the Telegram Application via a bot that has been added. In the “System A”, after photos are taken by the webcam using the OpenCV library, the image will be directly analyzed by TFLite based on the AI model used.

Whereas on the “System B”, images must be uploaded to the GCP Cloud Server first using the API from Google. The image will be analyzed using the Google Cloud Vision API. Then, the Cloud Vision API will send the analysis results back to Smart-Basket.

After the two systems successfully analyze the object. The results of the analysis will be implemented and obtained a shopping receipt. The shopping receipt is then sent to end-users using the Telegram application with the Telegram API interface.

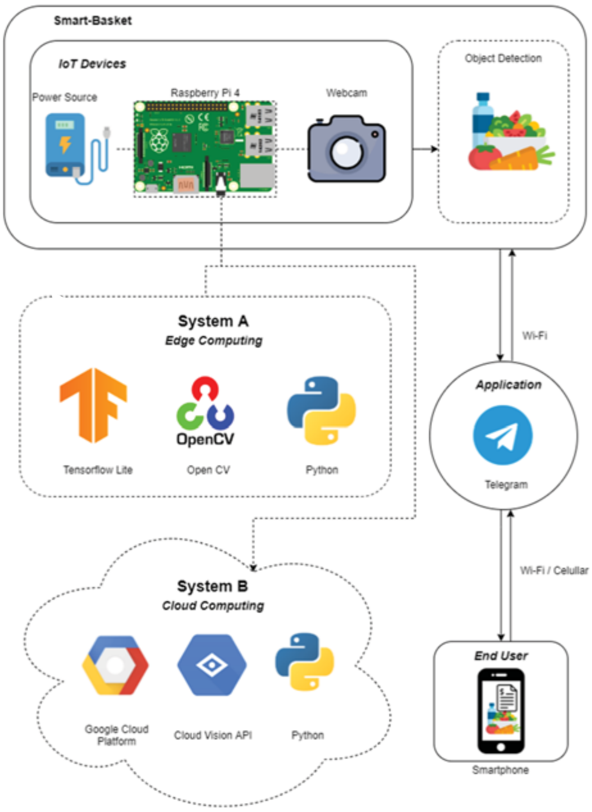



Fig 2. Design as Developed by Author

The following are the steps on designing the system and conducting the testing of this system.

Step 1. *Hardware and Software*

Connect Raspberry Pi 4 device with webcam and power source. Install Raspbian OS Buster to Raspberry Pi 4 as operating system and Open CV for object detection software. Next, Install the TFLite software for the "System A" system and install the Google Cloud Vision API software for the Cloud "System B".

Step 2. *Coding*

Integrating software and library such as Open CV library, Telegram API, and TFLite library for "System A". And for the "System B", instead of using TFLite Library as an AI program, the system uses Google Cloud Vision API that has been provided by Google AutoML. Both systems use Python programming language as the base of the main program.

Step 3. *System Testing and Data*

"System A" and "System B" are tested in LOS and NLOS conditions with a resolution of 1280x720 pixels. The data obtained in the form of the processing time for each system and the RSSI value for each condition. Fig. 3 and Fig. 4 shows our proposed devices when conducting the system testing.

Step 4. *Data Analysis*

Analyzing performance between two systems that have been tested by examining the processing time for each system and comparing the RSSI values in each condition.

## IV. Results And Analysis

After building the device is complete, the next step is to test and measure the device's performance. Measurements are made by pulling the device from the reference point (router) to the specified endpoint. Testing condition is divided into two conditions: Line of Sight (LOS) and Non-Line of Sight (NLOS). Measurement were made 10 times from a distance of 0.1 meters, 0.3 meters, 0.6 meters, 1 meters, 1.3 meters, 1.6 meters, 2 meters, 2.3 meters, 2.6 meters, and 3 meters and with different types of condition to see the effect of distance and condition for each system. The data obtained are in the form of time performance and RSSI values.

### A. Time Performance

When shopping at the supermarket, there is a possibility that the device is not blocked and will be blocked by objects such as high shelves. We conduct two test conditions, namely the Line of Sight (LOS) condition and the Non-Line of Sight (NLOS) condition.

#### 1) Time Performance on Line of Sight (LOS) Conditon

LOS condition is sending and receiving signals that are not obstructed by the object (Without Obstacles). As shown in Fig.5, during the LOS condition, the "System B" system experienced significant fluctuations. Fluctuations in this graph indicate the system is not stable enough. When compared with "System A" graph, the graph is very stable with minimal distortion. The average time performance on the "System A" is 1.74 seconds, while the "System B" has an average of 10.46 seconds. It means that "Edge Computing" system is more stable than "Cloud Computing" system. This is because the "Edge Computing" system is not affected by connection stability and signal quality in the object detection process, whereas "Cloud Computing" systems must first upload the images to the cloud server, which requires a stable connection and good signal quality.

Fig 3. The Testing of Proposed IoT-Smart Basket

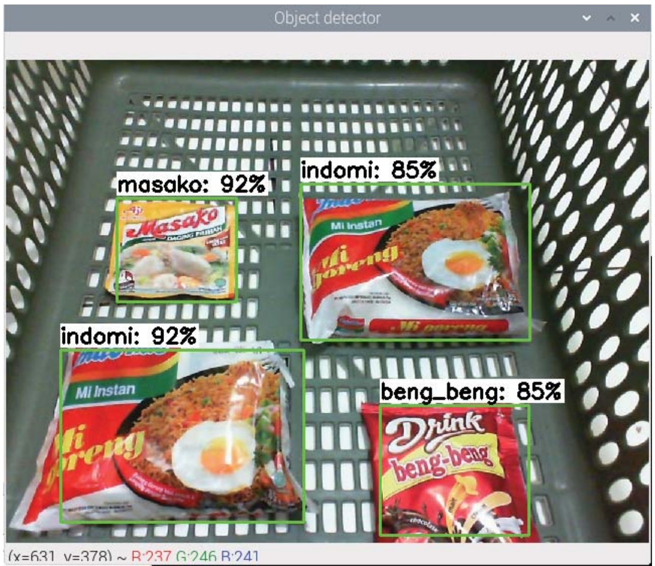



Fig 4. The Testing of Object Detection

#### 2) Time Performance Non-Line of Sight (NLOS) Condition

NLOS condition is sending and receiving signals blocked by certain objects (with Obstacles). As shown in Fig. 6, during the NLOS condition, "System B" has increased the graph twice, and the graph also experienced a slight fluctuation indicating an unstable system. Then, on "System A", the graph is very stable, there is no large data variation. The average time performance on the "System A" is 1.75 seconds while the "System B" has an average of 5.36 seconds. The test indicates that Smart-Basket utilizing "Edge Computing" System is more stable than "Cloud Computing" system for the condition of NLOS. Same as the previous condition that the "Edge Computing" system is not affected by connection stability and signal quality in the object detection process.

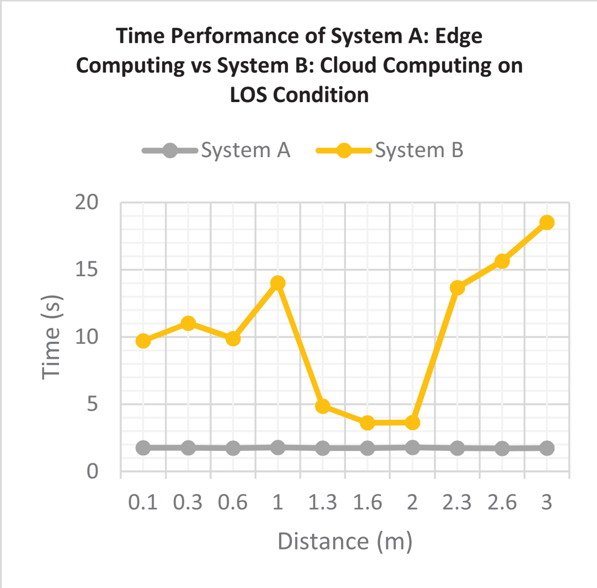



Fig 5. Time Performance of "System A" and "System B" on the LOS Condition

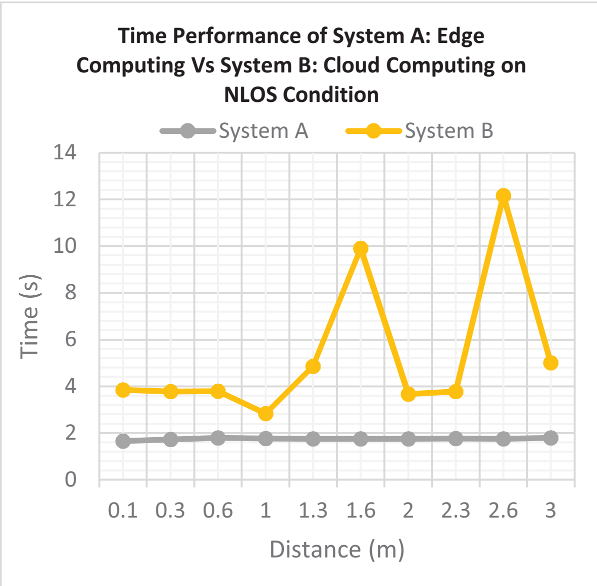



Fig 6. Time Performance of "System A" and "System B" on the NLOS Condition

### *B. RSSI Value*

Like the test conditions on Time Performance, testing on RSSI values can also be affecte by Line of Sight (LOS) and Non-Line of Sight (NLOS) conditions.

#### *1) RSSI Value on Line of Sight (LOS) Condition*

In the LOS condition, Fig. 7, when Smart-Basket moves away from the test point, both systems experience a decrease in RSSI values. System A has a maximum RSSI value of -29 dBm and a minimum of -55 dBm. While System B has a maximum RSSI value of -18 dBm and a minimum value of -49 dBm. "Edge Computing" and "Cloud Computing" system have both decreased in RSSI value. This indicates that RSSI value depends on the distance between the device and the test point, which is the source of the signal generated.

#### *2) RSSI Value on Non-Line of Sight (NLOS) Condition*

In the NLOS condition, Fig.8, when the Smart-Basket moves away from the test point, both systems experience a decrease in RSSI values. System A has a maximum RSSI value of -49 dBm and a minimum of -75 dBm. While System B has a maximum RSSI value of -51 dBm and a minimum value of -72 dBm. In NLOS condition, Edge Computing and Cloud Computing system have lower RSSI values compared to LOS condition because both devices are blocked by objects which will block the signal transmitted by the test point. As in the previous condition, both systems experience a decrease in RSSI value because of the increase distance between the devices and the test point.

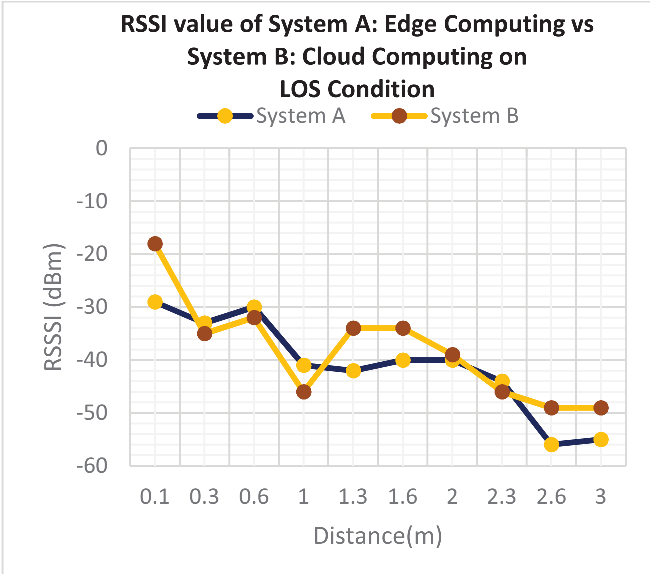



Fig 7. RSSI Value of "System A" and "System B" on the LOS Condition

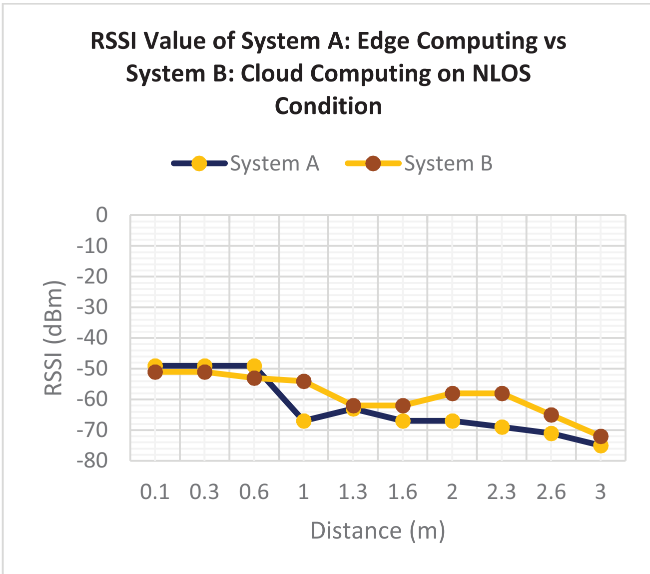



Fig 8. RSSI Value of "System A" and "System B" on the NLOS Condition

## V. CONCLUSION AND FUTURE WORK

This paper has proposed the system to build the best system for IoT-Smart-basket with comparing the performance between "Edge Computing" system and "Cloud Computing" system. The system consists of Raspberry Pi hardware and webcam. Python, TFLite, OpenCV, and

Google Cloud Vision API software to detect shopping objects. The object detection results are calculated and sent to end-users in the form of shopping receipts through the Telegram application. The result of the comparison between the two systems is that the System A: "Edge Computing" is more stable than the System B: "Cloud Computing" in all condition. The System A: "Edge Computing" is also not affected by RSSI values and the distance of the device to the test point. However, the Edge Computing system has a limitation that it requires more demanding hardware and space in the device for running the complex multi-object detection. The future work is to build and test the proposed design the in a real system scenarios and scalable systems.

## VI. Acknowledgments

This paper is supported by Hibah PUTI 2020 Universitas Indonesia. Mr. Nandiwardhana's bachelor project work is supervised by Dr. Muhammad Suryanegara (suryanegara@gmail.com).

## References

[1] K.Lalitha, M. Ismail, S. K. Gurumurthy and A. Tejaswi, " DESIGN OF AN INTELLIGENT SHOPPING BASKET USING IoT," *International Journal of Pure and Applied Mathematics,* vol. 1, no. 151, 2017.

[2] S. Agarwal and B. Tiwari, "SMART JAR: AI TECHNOLOGY BASED GROCERY SHOPPING," *International Journal of Electronics and Communication Engineering and Technology (IJECET),* vol. 9, no. 3, pp. 47-52, 2018.

[3] M. Chiang and T. Zhang, "Fog and IoT: An Overview of Research Opportunities," *IEEE Internet of Things*, pp. 1-12, 2016.

[4] Karjol, Srinidhi & Holla, Anusha & c b, Abhilash. (2018). An IOT Based Smart Shopping Cart for Smart Shopping. 10.1007/978-981-10-9059-2_33.

[5] Medaglia, Carlo & Moroni, Alice & Volpi, Valentina & Ceipidor, Ugo & Vilmos, Andras & Benyó, Balázs. (2011). Services, Use Cases and Future Challenges for Near Field Communication: the StoLPaN Project. 10.5772/20358.

[6] P. Chandrasekar and T. Sangeetha, "Smart shopping cart with automatic billing system through RFID and ZigBee," *International Conference on Information Communication and Embedded Systems (ICICES2014)*, Chennai, 2014, pp. 1-4, doi: 10.1109/ICICES.2014.7033996.3.

[7] S. Mekruksavanich, "The Smart Shopping Basket Based on IoT Applications," *2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China,* 2019, pp. 714-717, doi: 10.1109/ICSESS47205.2019.9040750.

[8] Masood, Aniq. "Recognition of Grocery Items in a Shopping Cart.", *Stanford University*. 2015.

[9] Oracle, "What Is the Internet of Things (IoT)?," Oracle, [Online]. Available: https://www.oracle.com/internet-of-things/what-is-iot.html. [Accessed March 2020].

[10] Isyanto, Haris & Arifin, Ajib & Suryanegara, Muhammad. (2020). Design and Implementation of IoT-Based Smart Home Voice Commands for disabled people using Google Assistant. 1-6. 10.1109/ICoSTA48221.2020.1570613925.

[11] Hayati, Nur & Suryanegara, Muhammad. (2017). The IoT LoRa system design for tracking and monitoring patient with mental disorder. 135-139. 10.1109/COMNETSAT.2017.8263587.

[12] i-scoop, "The Internet of Things (IoT) – essential IoT business guide," i-scoop, May 2017. [Online]. Available: https://www.i-scoop.eu/internet-of-things-guide/. [Accessed 28 4 2020].

[13] C. Chang, S. Srirama and R. Buyya, "Mobile cloud business process management system for the Internet of Things: A survey," Mobile cloud business process management system for the Internet of Things: A survey, vol. 70, p. ACM Computing, 2016..

[14] P. Sethi and S. R. Sarangi, "Internet of Things: Architectures, Protocols, and Applications," *Journal of Electrical and Computer Engineering,* vol. 2017, pp. 1-25, 2017

[15] W. SHI, G. PALLIS and Z. XU, "Edge Computing," *Scanning the Issue,* vol. 107, no. 8, pp. 1474-1481, 2019.