Secured Integration of IoT and Cloud Computing

Sudip Saha

Student Id: 011151314

Soumitra Saha

Student Id: 011153058

A thesis in the Department of Computer Science and Engineering presented in partial fulfillment of the requirements for the Degree of Bachelor of Science in Computer Science and Engineering

United International University

Dhaka, Bangladesh

January 2019

Declaration

We, Sudip Saha, Soumitra Saha, declare that this thesis titled, "**Secured Integration of IoT and Cloud Computing**" and the work presented in it are our own. I confirm that:

- This work was done wholly or mainly while in candidature for a [BSc] degree at United International University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at United International University or any other institution, this has been clearly stated.
- Where we have consulted the published work of others, this is always clearly attributed.
- Where we have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely our own work.
- We have acknowledged all main sources of help.
- Where the thesis is based on work done by us jointly with others, we have made clear exactly what was done by others and what we have contributed by ourselves.

____

Name: Sudip Saha, ID: 011151314 and Department: CSE

____

Name: Soumitra Saha, ID: 011153058 and Department: CSE

Certificate

I do hereby declare that the research works embodied in this thesis entitled "**Secured Integration of IoT and Cloud Computing**" is the outcome of an original work carried out by Sudip Saha and Soumitra Saha under my supervision.

I further certify that the dissertation meets the requirements and the standard for the degree of [BSc] in Computer Science and Engineering.

Dr. Salekul Islam
Professor & Head
Department of Computer Science & Engineering
United International University

Abstract

Over the recent years with the enhancement of smart application, IoT has changes the modern life in large-scale. Device which are used as Internet of Things (IoT) are normally low prices, consumed low power and the computational cost is not as higher. In this circumstance using an IoT device for a small organization as well as ordinary people is very much spontaneous. On the other hand, if you are a burdensome user in terms of internet then cloud server is probably your first choice because all the computational complexity is managed in the background server. In this paper we actually compare how to handle both the IoT device and cloud server and as well as how continuity is being changed with respect to device, place and position of device. We use face recognition as application to build a relationship between edge cloud and core cloud with respect to time by using an IoT device (Raspberry Pi). With the use of MQTT (Message Queuing Telemetry Transport) protocol we also maintain the secrecy as well. As a result of using huge amount of packet with a specific protocol and face recognition as application to store image, recognize the image and detect the image from the image dataset, the latency of the network may be increased as well. More data you transferred, more complexity you must be faced. Transfer a huge amount of packet in core cloud is not much easy and for that we reduce the latency of network in the core cloud and edge cloud also. We evaluate the result for both edge cloud and core cloud with the use of IoT device and without also. The ultimate result will show you that if you use edge cloud with reference to core cloud, it will be faster and not time consuming.

Acknowledgement

It has been great experience for us to work with our supervisor, Dr. Salekul Islam. Throughout this thesis, we have earned vast of knowledge in the field of IoT, Cloud Computing, networking sectors and also developed our programming skill. We would like to thank him for being so kind, helpful and for revealing us to the real-world application. We also want to thank to our honorable faculty, Mr. Nasif Muslim, for helping us so much. We are also thankful to the developers of the open-source libraries that we have used in our application.

At the end, much honor for our family and friends for their clench and exhortation.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Motivation

Though Cloud computing and IoT are two miscellaneous topics and both of them have their own formation as well as specific terminology but they are very much interrelated in our day to day life. Integration of Cloud and IoT is known as Cloud-IoT illustration. We expected to shatter those current and future Internet by integrating these two technologies which we can call Cloud-IoT.

In research, industry or education, integrating Cloud and IoT is really an important, interesting and promising topics. We definitely analyze and discuss about the importance of integration of Cloud and IoT, what type of challenges we face to integrate them and how we can encounter them in real time. In order to integrate cloud and Internet of Thing we can easily detect the common features and also the benefits of integration.

Everything is not affordable for someone and the aggregate is not free for everyone. If you are an internet user, then you must want to buy everything for it. When it comes into IoT device and cloud server then it will be not much easy for somebody to afford. But if it possible to use the open sources then it will be very much beneficial for us. In this paper we use most of things which is open sources. To integrate IoT and cloud server we use an IoT device which is not much costly, and we use azure virtual machine as cloud which is totally free as a student version and an open source protocol.

Normally if we transfer a text message via internet then it is visible. It can be very much easy for an intruder to catch up the transferred message and that's not good for both sender and receiver. The message contains your personal information or anything that's are very much confidential with respect to secrecy. As a result of security purposes, it is very much risky. For that you must need to transfer your message in a secure way. As a result of security purposes here we use a specific protocol term which is MQTT. First, we encode the whole text or string and then pass it through MQTT protocol. By this time if any intruder can take any chances to see the text file, he/she can't see this as it is encrypted. Your file is totally safe with is. When the text is transferred into the desired location then the receiver can decode the encrypted file and gather the information safely. By using the MQTT protocol the security build up is done quite comfortably.

Here we transfer image through MQTT protocol and as application we use face recognition. As we use IoT device and transfer image through a protocol into a remote server, it is very much time complexing and the result might move towards error. Otherwise instead of transferring the image file we can easily send any types of video file with any size and any other file as well. This also gives us the opportunity to transfer a file securely.

1.1.1 MQTT

In order to build up secured integration between IoT and Cloud Computing, we have used MQTT protocol which is a publish-subscribe-based messaging protocol. It works on top of the TCP protocol and it is designed for connections with remote locations.

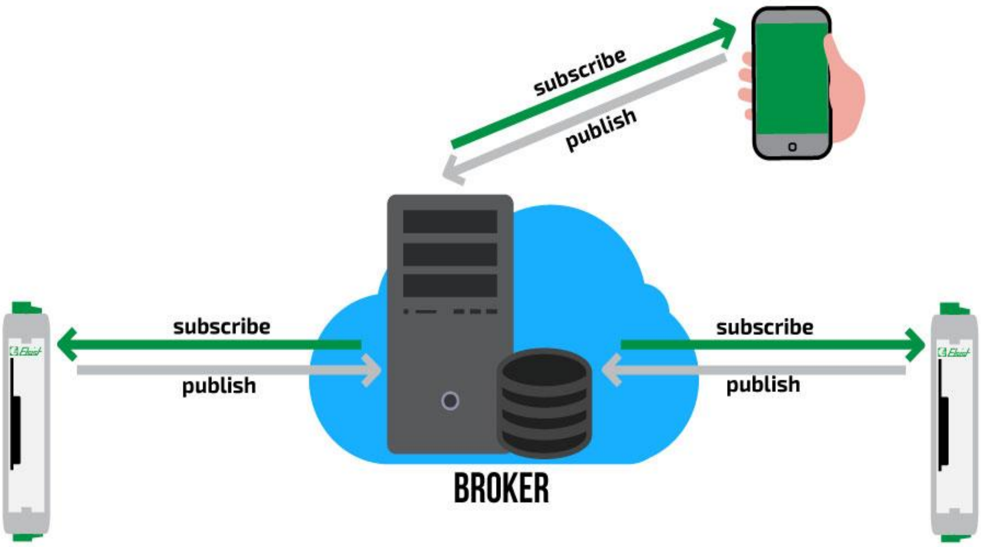

Figure 1: Workflows of MQTT Protocol

Despite others protocol, we have selected MQTT for its various usefulness: more efficient information distribution, increased scalability, a reduction in network bandwidth consumption dramatically, well suited for remote sensing and control, uses permission-based security. Others protocol like, Constrained Application Protocol uses a request communication pattern, and the Advanced Message Queuing Protocol, which, like MQTT, uses a publish/subscribe communication pattern but so much of useful as MQTT.

1.2 Objective

Here we use face recognition as application. In this application we use both private IP, public IP and cloud as well to transfer image through IoT device and private pc at the same time. Both use of edge cloud and core cloud and also a private cloud we can easily differentiate the result for further jurisprudence.

From the IoT device and the private pc we transfer the image file as encoded string. After receiving the encoded string in the cloud server or local server (which is used as a public IP server), we decode it and collect the image. By using OpenCV (Open source computer vision) as a library function we completed the image comparison.

We count the total time for face recognition of the selected image (sending time + receiving time), total time for face detection of the selected image (sending time + receiving time) and also count the time to train the whole image dataset that we used as database in both second and millisecond. Investigate the total detection time and recognition time in terms of image processing time and also the transferred data with total number of images in the image dataset. From that we can easily compare the minimum amount of time that we required to detect the selected image among the servers.

1.3 Organization

Remaining part of our thesis is organized as follows. In Chapter 2, we described about the main topics from previous literature – Internet of Things (IoT), Cloud Computing and Security Protocol. We use face recognition as application which is elaborated in chapter 3 and we have also said about different methods of it. In chapter 4, we have analyzed all result and included clear comparison tables. At last, in chapter 5 include the conclusion part of our thesis with specific result.

Chapter 2

Background and Literature Review

2.1 Internet of Things (IoT)

The Internet of Thing in short form IoT offering the end user with an unprecedented means of intercommunication with the web by using internet. In the telecommunications field which is growing rapidly, IoT provide a mammoth revolution which is mainly related to wireless communications. The Internet of Things (IoT) footstep is actually based on rational and self-configuring things which things is internally connected by the substructure of worldwide network.

We can easily estimate and investigate everything in the world and by doing this it is very possible for us to diminish the amount off cost, loss, waste, annihilation and so on [6]. By embedding intelligent components with the IoT device, we can easily bring our important things more smatter than before.

Things which is susceptible to connect with internet and gather information from different sources and also have the ability to process the collected information is called a IoT device [1] .The Internet of Thing (IoT) " is a network by which every types of physical objects, vehicles, buildings etc. are easily interconnected through different kind of software, electronic device, sensors which can communicate with internet to collect data and gather information for further use like searching data from large data set, pre-processing the data or sharing the data. The performance measurement of an IoT device is depends on how much it is feasible and also its security in terms of collecting data and transfer data [8].

To know about internet of things, first of all we need to know about the application of IoT devices [7]. A huge amount of application is visible for IoT devices like thermos tons, smart car, smart city, smart hospital, different types of electronic application, alarm clock for signs of the fire and so on.

Now, talking about how these applications can help you easily by using any IoT device. Suppose you are out of the house and will come home after some time. Now considering the weather, if the sun is shining then the windows of your house and the fan will be open or the fan as well as the window will remain close. How can you do this? With the help of some IoT devices using the Internet, you can do this easily. By using the same technology, you can easily handle a smart hospital or a car.

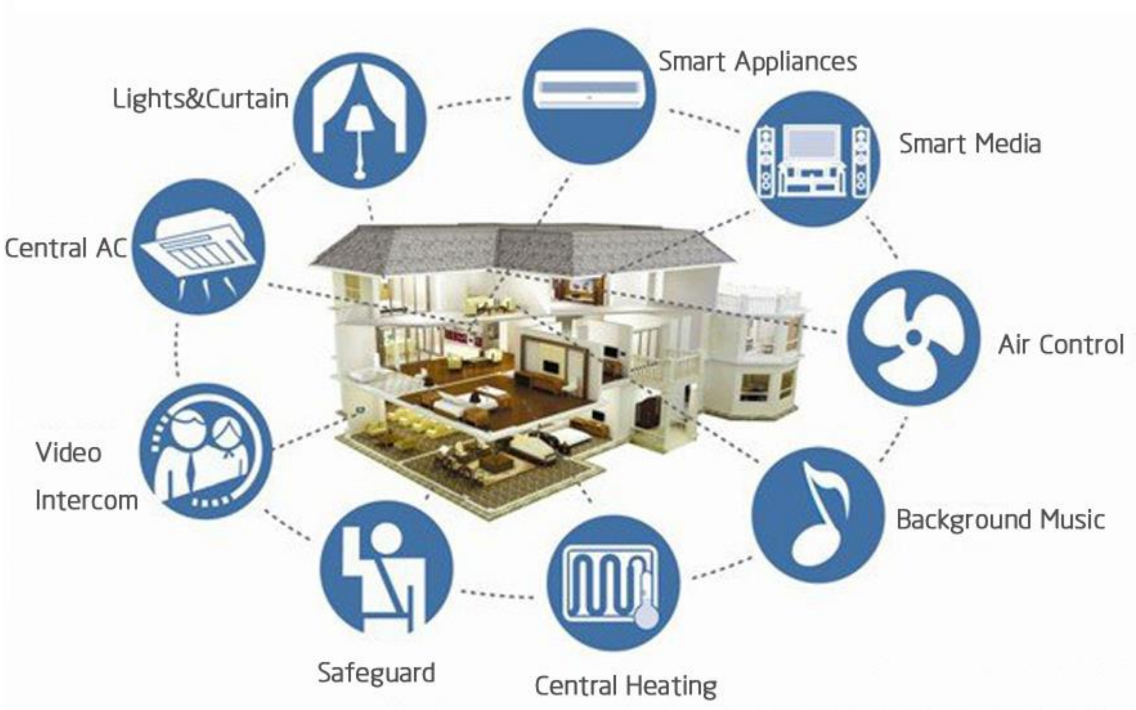

Figure 2: Smart Home [18]

This (Figure 1) is an example of IoT application. In this smart home, all above things can be controlled through internet. The greater portion in terms of picturesque effects about the Internet of Thing is that by using it one can easily optic his/her homey and working field at the same time.

2.2 Cloud Computing

Cloud computing is actually known as Internet-based computing or on demand internet computing which included different types of services like server with huge amount of storage and different types of application [9]. This type of facilities is delivered to the organizations because without any use of computer hard drive one can easily use data and store information through internet [10]. Servers are used to store information.

Edge cloud

After collecting any kind of valid data which is useful for over and above process, if the data processing part is controlled by the edge of a network rather than the cloud then it's called edge computing [3]. In edge cloud data capturing, data pre-processing for further process, data transmitting or receiving all are completely (happened in specific causes) or partially done by different types of smart devices [11]. This smart device is controlled by a central server to gather information from the devices and give decision. Routers and switches are the most common example of edge cloud [17]. From a specific IP address router take the instruction and give it to switch for sending it to the ISP provider.

Core cloud

When the application process of data is over past in cloud or the processing power of data is hold by the cloud but not in the edge of a network then it's called core cloud. As capturing, processing or sending data everything is completely done by cloud himself, no smart devices is needed here. It's uses as a central network and for that sub-network are interrelated with each other through core cloud. As everything is centralized, developing and controlling such a huge thing is tough but in terms of security and services core cloud is far better than edge cloud. Facebook and Google are the two famous example of core cloud.

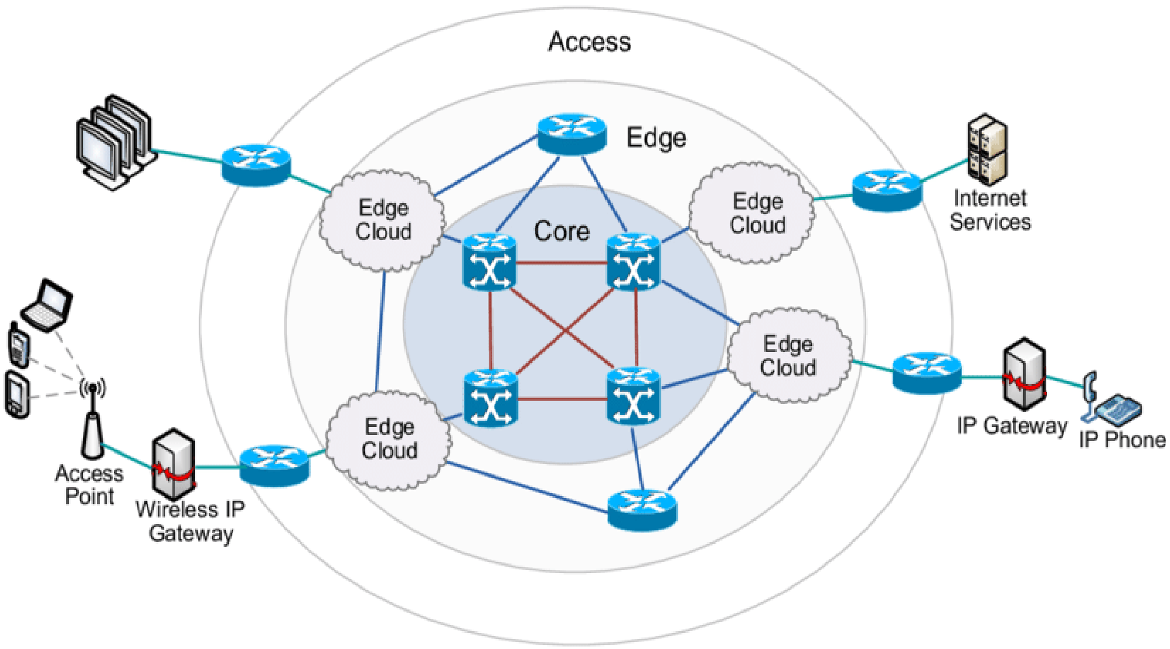

Figure 3: Edge Cloud and Core Cloud [19]

Optimization of cost, more flexibility, speed, performance, elasticity, efficiency, pay per use, more facility and resiliency are the main reason to use cloud instead of a computer device [12]. As the minimum infrastructure cost of a local server is too much high because of maintaining a server station, networking, storage, data center and so on, it's very much difficult for a low budget association. By using both software and hardware simultaneously, cloud computing can deliver a service which is very much beneficial with respect to a computer device in the growing world now a days [15]. Without having any types of local server or application one can easily go through by this.

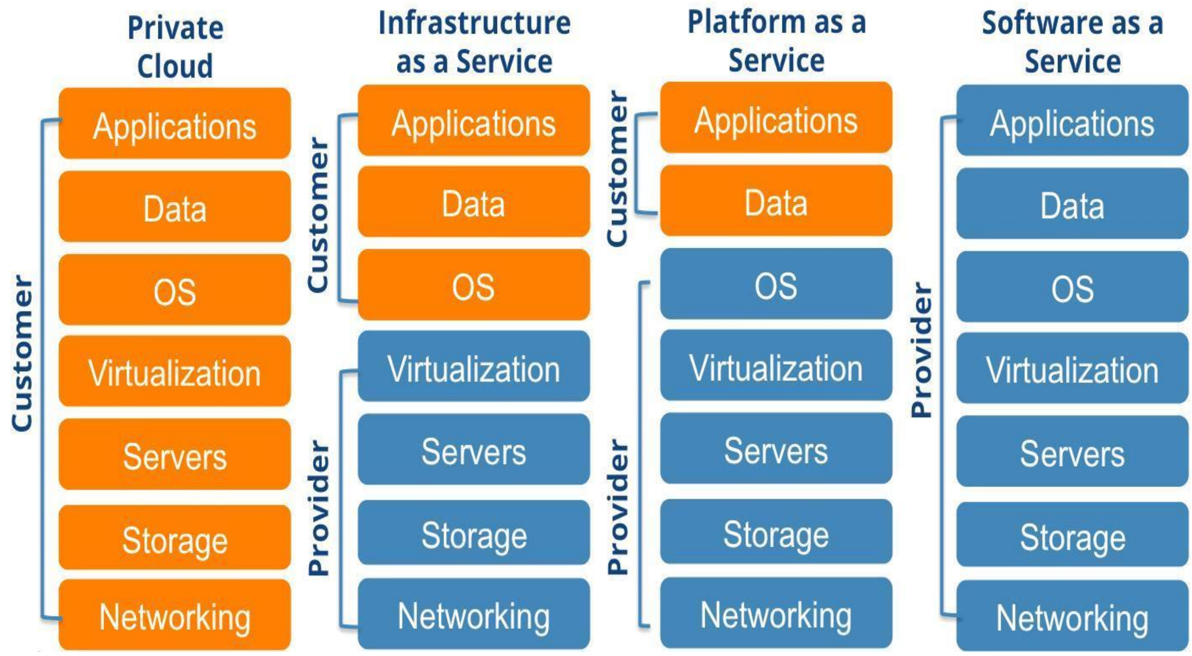

Figure 4: Different types of Cloud services [20]

With the help of cloud computing from wherever you are, you can easily access any software or application. Access of a cloud can gain easily from the third party who are the cloud service provider. Interact with cloud with the use of API is more reliable than anything in today's competitive world as internet is not reliable because of intruder [14]. A public cloud is more secure than a local server or personal computer, nevertheless.

2.3 Security Protocols

"Information is wealth". In the 21st century you are not safe at all with reference to internet. If you are a regular user of internet, then secrecy of internet is a must otherwise you will be in burdensome trouble. To collect one’s information through internet is not a big deal nowadays. When you in internet, you are always in an intimidation of attack. To be safe and secure regarding internet you needs to follow some specific provision.

To transfer or receive data from sender to receiver or vice versa with the help of electronic devices we use some specific rules and regulation which is called protocol. So, when we use this type of protocol for security purpose then it’s called security protocol. To secure the system layer security is first. As an example, for data integrity, data authenticity, data confidentiality and data availability network security are badly needed. To connect IoT devices through internet which are far away from each other, network security is must. If you transmit any data without security, then it will be disastrous for you notwithstanding. Some basic concept about network security and follow some specific protocol can assured you that you are totally free from some kind of massive attack like Denial-of-service (DoS) attack, distributed denial-of-service (DDoS) attack, Man-in-the-middle attack etc.

Chapter 3

Face Recognition

As an application, we have developed face recognition which is a method of identifying or verifying the identity of an individual using their face.

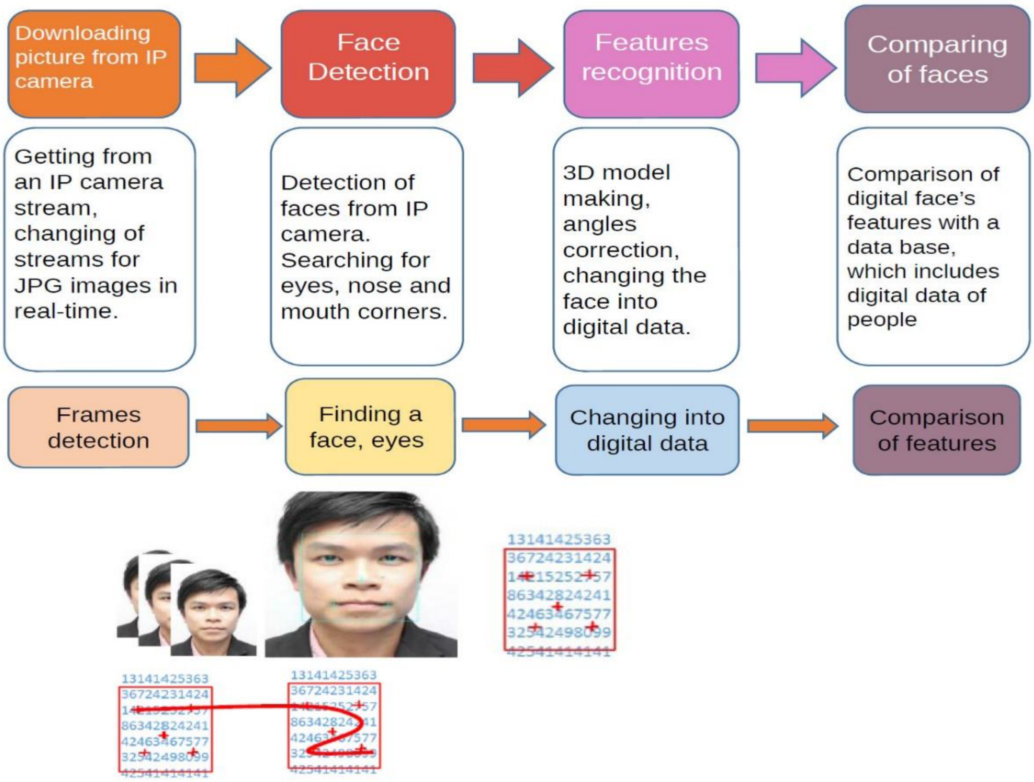

Figure 5: General process of Face Recognition [21]

In our thesis, we actually integrate IoT device and cloud server. As an IoT device we use raspberry pi and as edge cloud we use edge cloud as United International University data center and as core cloud we use Microsoft Azure virtual machine. The main countenance of our paper is to transfer image from an IoT device to remote cloud and then compare the transferred image with the image dataset which is already in the cloud by using face recognition algorithm [4].

Face recognition is a system where an image of a person is given. By collecting some kind of information, one needs to compare the image with the selected image and give you the decision. Actually, the whole process is done through face recognition and face detection and they follow different types of algorithm. Local Binary Patterns Histograms (LBPH) is one of the most famous face recognition algorithms and in our paper, we use it.

First work is to detect the face and by face detection one can easily detect the location of the image and the size of that image which is needed for further demonstration.

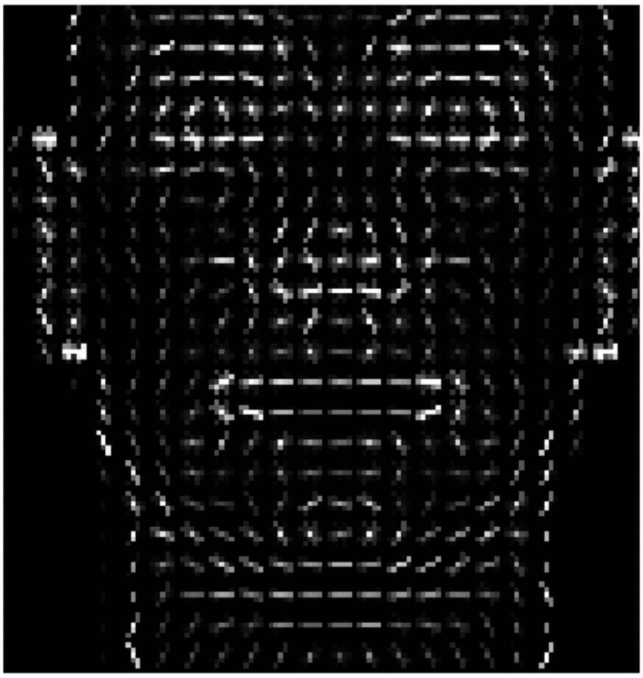

Figure 6: Histogram to detect the face [22]

When the detection part is completed then the authority is realized. We already have an image dataset and we need to train the dataset first [16]. For that we first set an identical name for each image in the dataset. As a result, the algorithm can recognize the name which is helpful for it to identify the image because each person may have different image and the input and also the given output is not same all the time.

The detected face which is already extracted and resized also, face recognition algorithm is used to be accumulated the peculiarity. The peculiarity is then described that from the image dataset which image has the equilibrium or equivalence to the transferred image by the IoT device.

For authentication of an image, every image in the existing dataset is compare with the selected image and the give a decision. It's basically calls one to one authentication. It's a binary patterns histogram and for that by comparing two image the algorithm will give you a result which helps you to take a decision that the image is matched or not. As LBPH uses one to one authentication, it has better chances to compile the maximal resemble image.

3.1 Generic Architecture

Our IoT application's architecture is described below by a flowchart. Here the client side and the cloud part are operating and communicating through MQTT protocol. In the client part every image is captured and save in a specific location. One image is transferred at a time in the cloud part as string content. The content is received by the cloud and extract it to previous image and then the image is stored. Recognition application is taking place in the cloud part. After completing the recognition and detection part if the image is matched with an image in the existing dataset then a notification is transferred to the exact location where the selected image is stored [18]. The notification is also transferred through the middleware MQTT protocol.

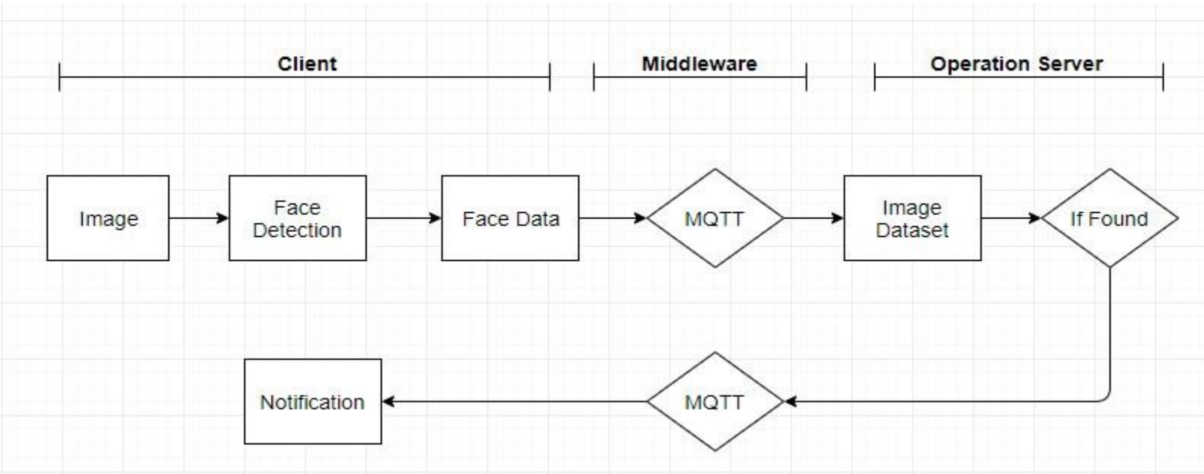

Figure 7: Flowchart of proposed architecture of Face Recognition System in IoT

3.4 Development

For the development of our process, first of all, in terms of security purposes we use MQTT protocol as a middleware. This is a special protocol which works over TCP layer. We use the MQTT protocol for both edge cloud and core cloud. As an IoT device we use raspberry pi device which helps us to transfer data through internet in the cloud.

3.4.1 IoT middleware: MQTT

MQTT is a simple messaging protocol which is designed for constrained devices with low-bandwidth. So, it's the perfect solution for Internet of Things applications. MQTT allows to send commands to control outputs, read and publish data from sensor nodes and much more.It can be utilized by clients based on TCP/IP. The standard port exposed by brokers is 1883, and it is not a secure port. Those brokers who support TLS/SSL typically use port 8883. For secure communication, the clients and the broker rely on digital certificates. AWS IoT is one of the secure implementations of MQTT, which requires the clients to use the X.509 certificates.

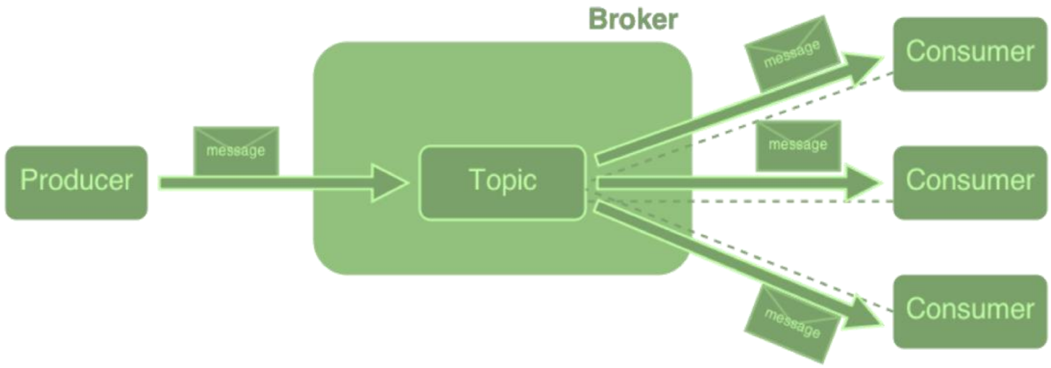

Figure 8: MQTT Broker [23]

3.4.2 Application in Broker part

Raspberry pi and pc both are used for client part in our thesis. For the development of cloud application, we used windows environment. As we demonstrate our work with the help of java programming language, we installed Oracle JDK 10.0.2, OpenCV, LBPH algorithm and face recognition for java. To describe the whole application process in the broker part two modules are described below.

Encoding module after capturing image

After capturing the image, first work is to encode the image as string content. We need to transfer the image into the cloud and for that first we encode it then transfer it with the help of a specific protocol which is described earlier. As only encoding is needed here, any kind of feature extraction or resize of the image is not an issue.

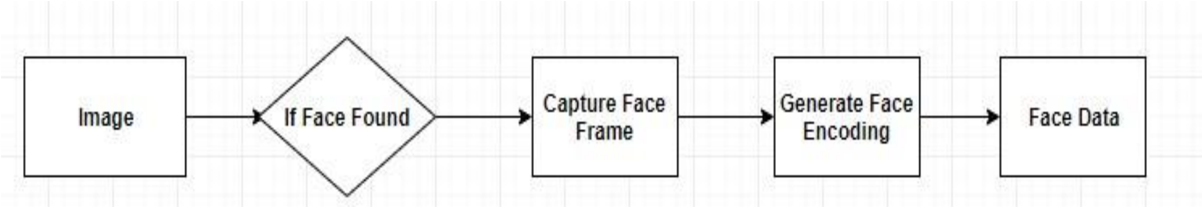

Figure 9: Flow diagram of Detection and Encode Module

Data transfer module

In this module data transfer is take place. After encoded the desired image now we need to transfer the image into the exact location. For transferring the data, we use a protocol. We use raw java code to transfer it. The encoded data is transferred in the cloud with security. In the cloud part, we can't store the encoded string. For that in the cloud part, we decode the image first and then store into a location to compare it with the previously stored data set.

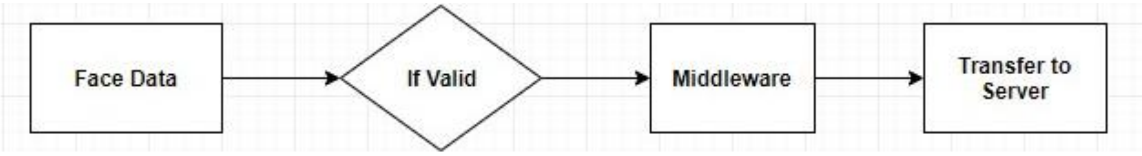

Figure 10: Flow diagram of Data Transfer Module

3.4.3 Server-side application

The application is developed in java. As ide we use eclipse here. Here we train a dataset with 600 images. First of all, we recognize the image that is receive from the IoT device. This is our desired image which we want to detect. As platform we use OpenCV and LBPH as algorithm. This algorithm selects a single image at a time and match it with the recognized image. A threshold value is set to detect the accuracy. When the accuracy is greater or equal to the predefined value then we conform that the image is matched. After that with the MQTT protocol we transferred a conformation notification that the image is matched with an image in the existing dataset. The notification message is transferred to that location where the actual receive image is located.

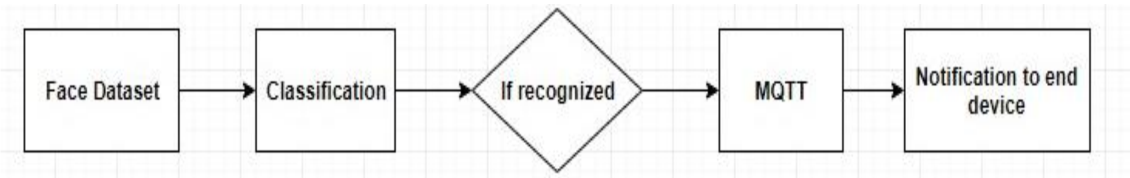

Figure 11: Flow diagram of server-side application

Chapter 4

Result and Analysis

4.1 Application Processes

- Send a request to recognize and upload an image
- In Sending applied MQTT Protocol
- In Broker and Subscriber, Face Recognition applied, used the OpenCV library.
- If Face Recognition process success, store that recognize image.
- We applied previous 1st step both using Raspberry Pi and without Raspberry Pi.

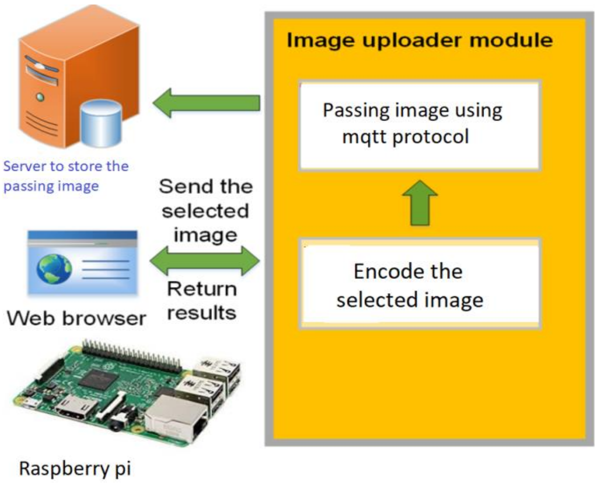

Figure 12: Application Process 1

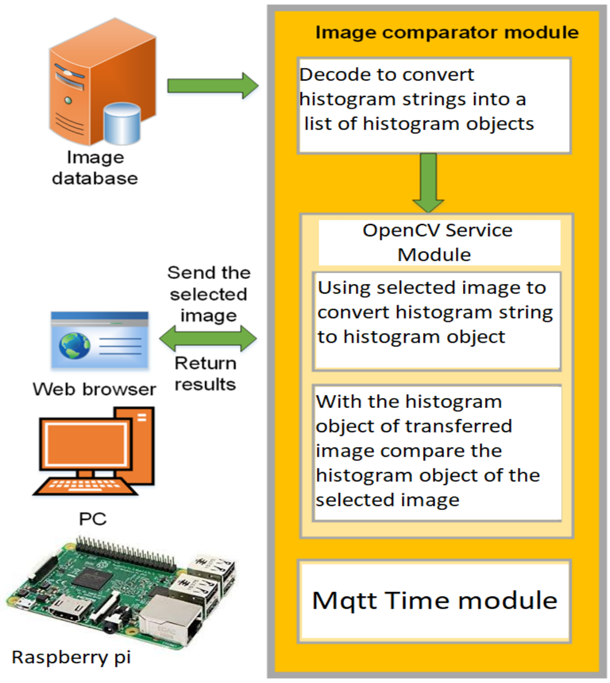

Figure 13: Application Process 2

4.2 Performance Evaluation

<u>Experimental setup</u>

At first, we have taken images, to perform face recognition. For this, we have built .xlsx file.

Table 1: Image database

Number of images in the image database	file size
606	21.9 MB

After image database processing, we have chosen our platform. We have considered three platforms. Each one's hardware specifications is different.

Table 2: Platform specifications

Types of cloud	Intranet	Edge cloud (Data center of UIU, Dhaka)	Core cloud (Microsoft Azure)
Processor	Intel(R) Core (TM) i3-4130 CPU @ 3.40GHz 3.40 GHz(2 Processors)	Intel(R) Xenon(R) CPU E5-2630 v3 @ 2.40GHz 2.40GHz (2 Processors)	2.3 GHz Intel XEON ® E5-2673 v4 (Broadwell)
RAM	8 GB	13 GB	3.5 GB
OS	Windows 10	Windows 10	Windows 10

For each platform, we have taken 2 different scenarios: one scenario with the IoT device Raspberry pi and another scenario without the IoT device Raspberry pi.

Table 3: Six Scenarios

Scenarios	Cloud	Is IoT Device used?
First Scenario	Edge cloud	Yes
Second Scenario	Edge cloud	No
Third Scenario	Core cloud	Yes
Fourth Scenario	Core cloud	No
Fifth Scenario	Intranet	Yes
Sixth Scenario	Intranet	No

Performance

Table 4: Data training time (Nano-second)

Environment	With raspberry pi in Broker	Without raspberry pi in Broker	With raspberry pi in Subscriber	Without raspberry pi in Subscriber
Edge Cloud	1.97390948E10	1.6455007E10	1.97390948E10	1.6383497E10
Intranet	9.045856506E9	1.1655962328E10	8.983209781E9	1.3350681825E10
Core Cloud	3.55485886E10	2.76986392E10	3.53285235E10	2.7628258E10

To train the entire dataset we count the total amount of time for both raspberry pi and pc. From the histogram given below we see that when we use intranet to store the dataset and train, it takes less time than the cloud. In core cloud it takes a huge amount of time to train the entire dataset at a time, but the result will be definitely better in intranet.

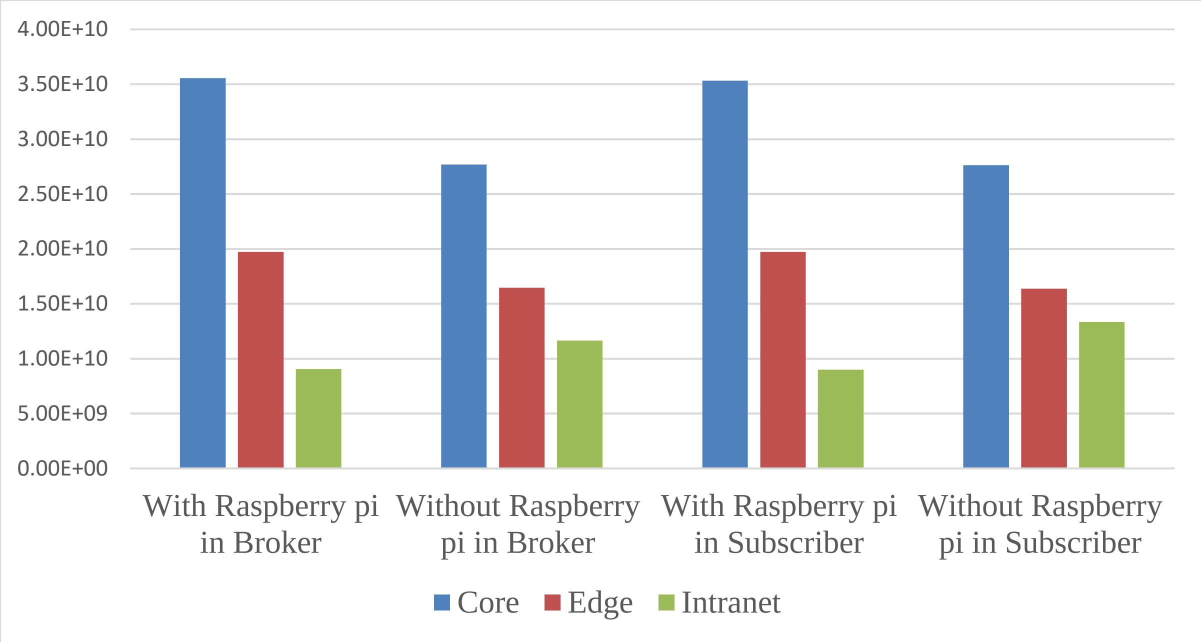

Figure 14: Histogram of Data training time

Table 7: In Broker, Recognition and Detection time (Nano-seconds)

Environment	Time to recognize the face	Time after recognizing face to reach file path in second	Total time for face recognition	Time for starting face detection to reach the dataset	After reaching, time for face detection from the dataset	Total time for face detection
In Intranet without Raspberry pi	2.516807E8	7500.0	2.516882E8	11800.0	5.362323E8	5.362441E8
In Intranet with Raspberry pi	1.824721E8	14200.0	1.824863E8	29900.0	5.772683E8	5.772982E8
In Edge Cloud without Raspberry pi	7.1801776E7	901.0	7.1802677E 7	4507.0	1.74955177E8	1.74959684E8
In Edge Cloud with Raspberry pi	5.346981E7	300.0	5.347011E7	3906.0	1.11510669E8	1.11514575E8
In Core Cloud without Raspberry pi	8.02312E7	1100.0	8.02323E7	12800.0	6.555166E8	6.555294E8
In Edge Cloud with Raspberry pi	3.219506E8	1300.0	3.219519E8	7700.0	4.745908E8	4.745985E8

Table 13: In Broker, total time (Nano-seconds) for face detection and recognition

In Core Cloud with Raspberry pi	In Core Cloud without Raspberry pi	For Public IP total time for face detection and recognition with	In Intranet without Raspberry pi	In Edge Cloud with Raspberry pi	In Edge Cloud without Raspberry pi
7.965504E8	7.357617E8	7.597845E8	7.879323E8	1.64984685E8	2.46762361E8

In broker, face detection time, face recognition time, total time for face recognition, total time for face detection and total time to detect and recognize the face is given. We calculate all the result with and without raspberry pi. From the histogram when we see the performance evaluation part, when we use cloud instead of intranet for both raspberry pi and pc it takes too long time to detect the image from the dataset, recognize the selected image which is transferred. The total time for face recognition and detection is also high which is not good at all in terms of computational cost and computational time. In this part the edge cloud and the core cloud almost take same amount of time. It takes 2.50E+08 NANO second in intranet which is at least three times better rather than edge cloud and core cloud.

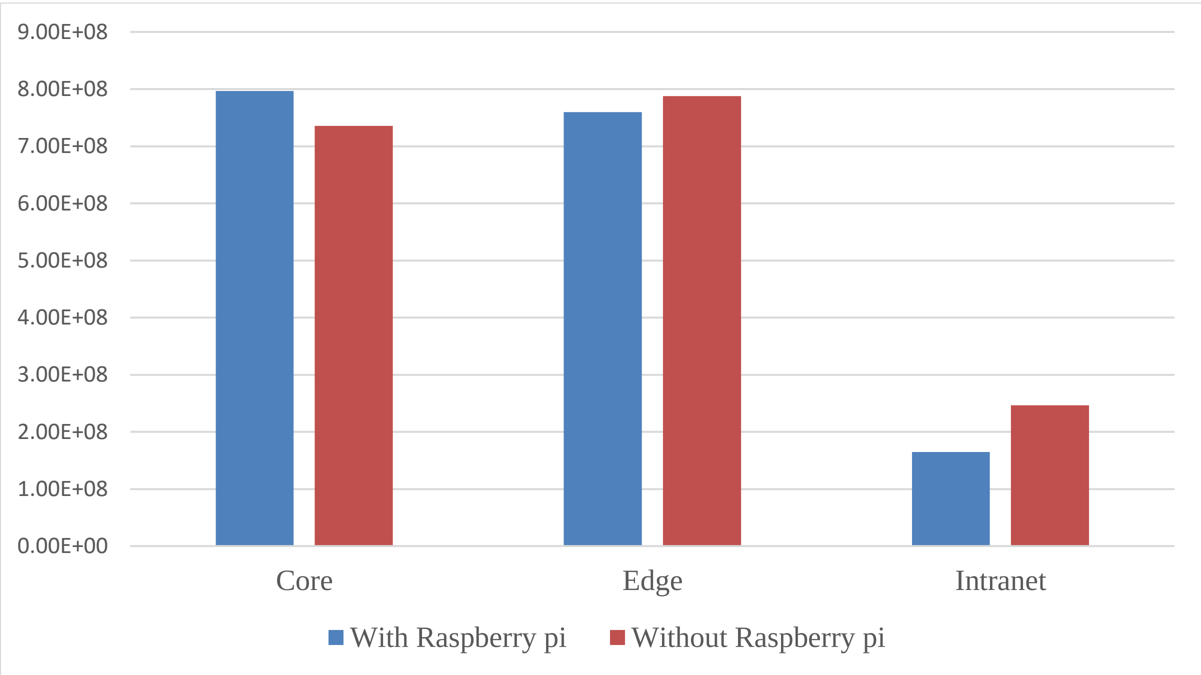

Figure 15: Histogram of Broker Performance Evaluation

Table 7: In Subscriber, Recognition and Detection time (Nano-seconds)

Environment	Time to recognize the face	Time after recognizing face to reach file path in second	Total time for face recognition	Time for starting face detection to reach dataset	After reaching, time for face detection from the dataset	Total time for face detection
In Intranet without Raspberry pi	2.710993E8	9400.0	2.711087E8	29300.0	5.779424E8	5.779717E8
In Intranet with Raspberry pi	1.824721E8	14200.0	1.824863E8	29900.0	5.772683E8	5.772982E8
In Edge Cloud without Raspberry pi	9.2209194E7	601.0	9.2209795E7	2705.0	2.45431065E8	2.4543377E 8
In Edge Cloud with Raspberry pi	5.3039426E7	300.0	5.3039726E7	2403.0	1.31102229E8	1.31104632 E8
In Core Cloud without Raspberry pi	9.8369E7	1500.0	9.83705E7	8200.0	5.274939E8	5.275021E8
In Edge Cloud with Raspberry pi	3.672266E8	1200.0	3.672278E8	7400.0	4.742384E8	4.742458E8

Table 20: In Subscriber, total time (Nano-seconds) for face detection and recognition

In Core Cloud with Raspberry pi	In Core Cloud without Raspberry pi	In Intranet with Raspberry pi	In Intranet without Raspberry pi	In Edge Cloud with Raspberry pi	In Edge Cloud without Raspberry pi
8.414736E8	6.258726E8	7.597845E8	8.490804E8	1.84144358E8	3.37643565E8

In subscriber part we see the same thing which is happened in the publisher part. For both raspberry pi and pc, it takes a lot of time to detect and recognize the image. As a result, the computational time is much higher in core cloud and edge cloud but it’s very much reasonable in intranet. It takes 3.00E+08 NANO second in intranet but takes three times more in cloud which is not good.

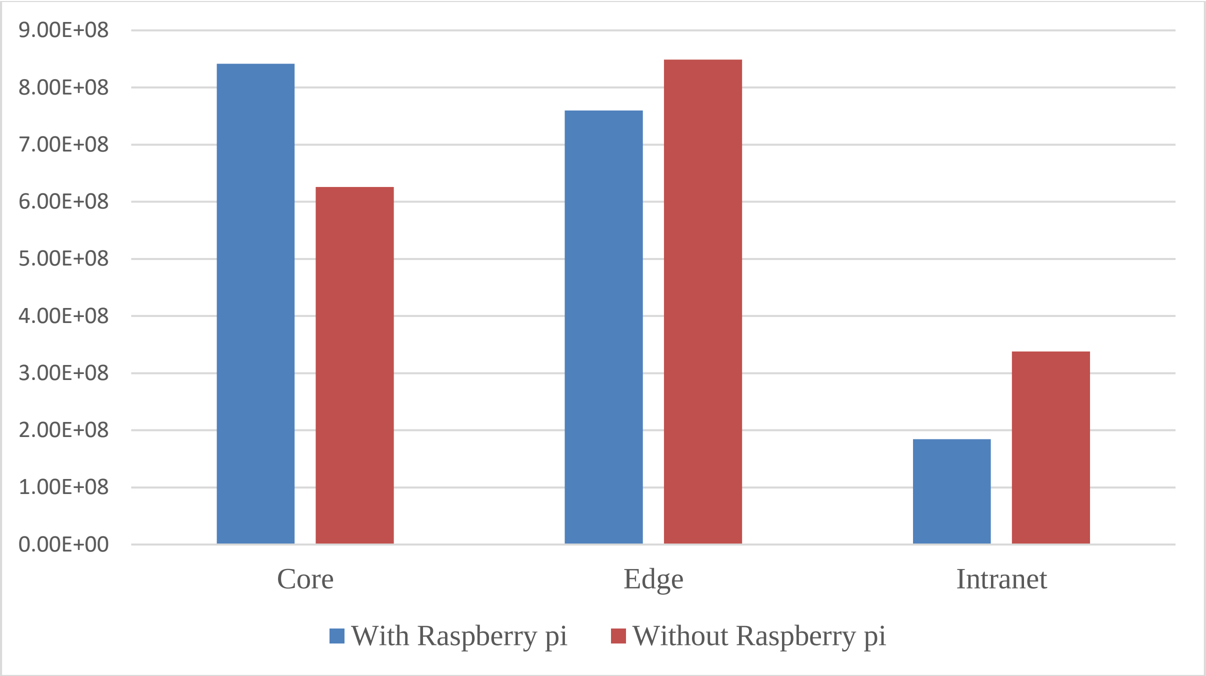

Figure 16: Histogram of Subscriber Performance Evaluation

Chapter 6

Conclusion

The main contributions of this thesis are as follows:

1. An infrastructure is constructed for face recognition by applying open source tools and integrating Internet of Things (IoT) and Cloud Computing.
2. Evaluate the contribution, achievement and representation of hardware and software device for both edge cloud and core cloud.
3. By using an IoT device (Raspberry Pi) and private pc we differentiate the result among the clouds in terms of the size of image dataset.

By using this distinguishable protocol with an IoT device we can easily deduce the network dormancy. It's visible that edge cloud consumes less time than core cloud (Microsoft Azure Virtual Machine). We also extract the size of image through a process. As a result, the total time of train the image dataset and also the processing time of the selected image is deduced.

Intranet always gives you better performance but as here we use both types of cloud with the help of IoT device and capture the performance for both, we definitely said that if we want to use an IoT device then we must need to use edge cloud instead of core cloud as edge cloud gives us better result with respect to time at least.

References

[1] S. Islam and J.-Ch. Grégoire, "Network Edge Intelligence for the Emerging NextGeneration Internet," Future Internet, vol. 2, no. 4, pp. 603–623, 2010.

[2] Salton, Kelvin, et al. "Face Recognition: Understanding LBPH Algorithm – Towards Data Science." Towards Data Science, 11 Nov. 2017, towardsdatascience.com/face-recognition-how-lbph-works-90ec258c3d6b.

[3] N. Muslim and S. Islam, "Face recognition in the Edge Cloud," Proc. Int. Conf. Imaging, Signal Process. Commun. - ICISPC 2017, pp. 5–9, 2017.

[4] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," Comput. Networks, vol. 54, no. 15, pp. 2787–2805, 2010.

[5] K. Ashton, "That Internet of Things thing," RFiD J., vol. 22, no. 7, pp. 97–114, 2009.

[6] H. Sundmaeker, P. Guillemin, P. Friess, S. Woelfflé, "Vision and challenges for realising the Internet of Things," Cluster of European Research Projects on the Internet of Things—CERP IoT, 2010.

[7] J. Gubbi, R. Buyya, M. Palaniswami, and S. Marusic, "Internet of Things (IoT): A vision, architectural elements, and future directions," Futur. Gener. Comput. Syst., vol. 29, no. 7, pp. 1645--1660, 2013.

[8] P. Mell and T. Grance, "Draft NIST working definition of Cloud computing," Referenced on June. 3rd, 2009 Online at http://csrc.nist.gov/groups/SNS/Cloudcomputing/index.html, 2009.

[9] M. Armbrust et al., "A view of Cloud computing," Commun. ACM, vol. 53, no. 4, pp. 50–58, 2010.

[10] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge Computing: Vision and Challenges," IEEE Internet Things J., vol. 3, no. 5, pp. 637–646, 2016.

[11] A. Greenberg et al., "The Cost of a Cloud: Research Problems in Data Center Networks," ACM SIGCOMM Computer Commun. Review, vol. 39, no. 1, pp. 68–73, Jan. 2009.

[12] E. Cuervo et al., "MAUI: Making Smartphones Last Longer with Code Offload," Proc. 8th ACM MobiSys, 2010.

[13] Manish Bansal. “Facial Recognition Using OpenCV in Java – FreeCodeCamp.org.” *FreeCodeCamp.org*, FreeCodeCamp.org, 25 July 2018, medium.freecodecamp.org/facial-recognition-using-opencv-in-java-92fa40c22f62.

[14] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog computing and its role in the Internet of things,” in workshop on Mobile Cloud computing. ACM, 2012.

[15] V. Rijmenam, M., “Self-driving cars will create 2 petabytes of data, what are the big data opportunities for the car industry,” 2017.

[16] S. Islam and J.-Ch. Grégoire, “Giving users an edge: A flexible Cloud model and its application for multimedia,” Future Generation Computer System, vol. 28, no. 6, pp. 823–832, 2012.

[17] X. Zhu and D. Ramanan, “Face Detection, Pose Estimation, and Landmark Localization in the Wild,” Comput. Vis. Pattern Recognit. (CVPR), 2012 IEEE Conf., pp. 2879–2886, 2012.

[18] “Technology & Home Improvement; From HVAC to Roofing to Windows to Home Security Systems.” American News Report, 17 June 2014, americannewsreport.com/technology-from-hvac-to-roofing-to-windows-to-home- security-systems-8820855.

[19] H. Chang, A. Hari, S. Mukherjee, and T. Lakshman, “Bringing the Cloud to the edge,” in Proc. IEEE Conf. on Computer Communications Workshops (INFOCOM WKSHPS), Toronto, ON, Canada, May 2014.

[20] Fung, Han Ping. "What Are Attacks That Can Occur on Cloud Computing Service Models". *ResearchGate*, 2018, researchgate.net/post/What_are_attacks_that_can_occur_on_Cloud_computing_service_models.

[21] "Face Detection and Recognition Systems". *Polgard.Com*, 2019, polgard.com/en/products/face-detection-and-recognition-systems.html.

[22] Rojas, Mario Alejandro. "Histogram of Oriented Gradients of The Face.". *ResearchGate*, 2011, ResearchGate.net/figure/Histogram-of-Oriented-Gradients-of-the-face_fig4_51586877.

[23] MSV, Janakiram. “Get to Know MQTT: The Messaging Protocol for the Internet of Things.” The New Stack, 22 Apr. 2016, thenewstack.io/mqtt-protocol-iot/.

[24] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, “The case for VM-based cloudlets in mobile computing,” IEEE Pervasive Comput., vol. 8, no. 4, pp. 14–23, Oct./Dec. 2009.