This is a repository copy of *Impact of Video Streaming Quality on Bandwidth in Humanoid Robot NAO Connected to the Cloud*.

White Rose Research Online URL for this paper:
https://eprints.whiterose.ac.uk/id/eprint/177681/

Version: Accepted Version

Conference or Workshop Item:

Aagela, Hamza, Holmes, Violeta, Dhimish, Mahmoud et al. (1 more author) (2017) Impact of Video Streaming Quality on Bandwidth in Humanoid Robot NAO Connected to the Cloud. In: Proceedings of the Second International Conference on Internet of things, Data and Cloud Computing, 22-23 Mar 2017.

Impact of Video Streaming Quality on Bandwidth in Humanoid Robot NAO Connected to the Cloud

H. **Aagela**
School of Computing and Engineering
Univesity of Huddersfield
United Kingdom
Hamza.aagela@hud.ac.uk.

V. **Holmes**
School of Computing and Engineering
Univesity of Huddersfield
United Kingdom
v.holmes@hud.ac.uk

M.**Dhimish**
School of Computing and Engineering
Univesity of Huddersfield
United Kingdom
Mahmoud.Dhimish2@hud.ac.uk

D. **Wilson**
Engineering School of Computing and
Univesity of Huddersfield
United Kingdom
d.r.wilson@hud.ac.uk

ABSTRACT

This paper investigates the impact of video streaming quality on bandwidth consumption during the transfer of video data from a humanoid robot 'NAO' to computing devices, used to perform face recognition tasks, and to the cloud. It presents the results of profiling the network performance of connecting NAO with an edge controller, and discusses the effect of using different qualities of video streaming on the consumed up-link bandwidth. This study considers the limitation of the up-link bandwidth in the Wi-Fi network. It compares the performances of Wi-Fi and Ethernet connections between the NAO robot and a computer. In addition, it examines the accuracy of the face recognition tasks using various streaming scenarios, such as colored video and black & white video. It investigates real-time video streaming using a wide range of frame rates, and video qualities, and their impact on the bandwidth, and accuracy of face identification. The results of our investigations are used to determine the acceptable video quality, frame rate, buffering and bandwidth that would give optimal results in face recognition using NAO robot, and enable efficient data transfer to the cloud.

CCS CONCEPTS

• Networks~Network performance analysis • Computer systems organization~Robotics.

KEYWORDS

Humanoid-robot, Wi-Fi, NAO, real-time video streaming, Face recognition, IoT, cloud

1   INTRODUCTION

Recently, developments in the hardware and software in robot technologies have increased dramatically, which has inspired many researchers to develop humanoid mobile robots, which are thus becoming more intelligent and usable in our daily life [1], such as NAO robot. However, in various aspects the robots are still limited, especially mobile robots regarding computational power and memory consumption. Therefore, this increases the need for offloading the computational process to a cloud system. The trend of the research points toward the humanoid robot has been evolved to be more socially active [16]. By analyzing the environment to gather information, for example, recognizing faces and objects, and using machine learning techniques, the robots could act autonomously. Therefore, the human behaviors and humanoid-robot interactions require to be defined correctly. The focus on robot design has been changed from stand-alone robot to a group of robots that share knowledge and resources [2]. Thus, robot networks [3] and cloud robotics [4] have been introduced as a solution to tackle mobile robot limitation by providing the robot with extra storage and high computational capability from the cloud. However, there are some challenges (e.g., limitation in the communication), where the up-link bandwidth is still restricted. In a remote environment computation model that is used to connect the

robot to the compute node, there is a need to send all the data from robot sensors and cameras over the network to manipulate the robot by the computer node.

In the cloud robotics implementation or any IoT implementation, the real-time communication [5, 13, 15] would be essential to obtain the offloaded sensor data, where any latency in the network or in the processing time has a negative impact on the robot performance. There are several factors that could cause an increase in the network latency, such as communication overhead and low bandwidth. However, it is vital to understand the required image quality and number of frames per second to be sent over the network to obtain the most efficient use of the available bandwidth.

Hence we analyzed the network performance with four different video qualities in various conditions such as, color and black & white and changing the number of frames, in order to define the lowest video quality that could be used to offload video data to the cloud This analysis will help to define the relation between the video quality and the consumed bandwidth, and an impact on the accuracy of a face recognition due to the delay or buffering in high quality video streaming within the receiver side.

The main challenges come from handling large volume of data characteristically associated with video streaming [17], robot sensors and information visualization (3D maps), and the intention of reducing latency and allowing functioning control and manipulation in the compute node [6]. One of the possible solutions is the application of cloud computing technology. A cloud controller with real-time communication, such as live video streaming, is required to provide visualization in the cloud robotic system. If the correct video quality and number of frames per second (fps) is selected, this would reduce the latency caused by data buffering to obtain smoother video streams. This buffering is the key reason for the video delay, which leads to a delay in the response of the robot operations in the interface of the controller.

The main motivation for this study is linked to the idea of implementing the cloud robotic communication to connect the NAO robot with Virtual Machines (VM) running on an Azure cloud [15]. This will require a high bandwidth to offload all sensor data of the robot to the cloud. Therefore, the robot uplink bandwidth consumption will be an essential aspect to be considered. Finding out the minimum acceptable video quality and frame rate that consumes less bandwidth, causes less buffering and still provides the end user with a clear picture is necessary to provide a high face recognition accuracy rate.

The rest of the paper is organized as follows: Section 2 presents the findings from related work; Section 3 outlines the specification of the robot used in the experiments. Sections 4 and 5 define the software environment and the experimental setup and the results, and Section 6 summarizes our findings and recommendation for future work.

2 RELATED WORK

There are several studies that consider the efficient real-time communication between the robot and the compute node, and the challenge of sending a high bandwidth sensor and camera data over the network. The Robot Web Tool RWT is an open source tool that utilizes the Robot Operating System (ROS) [18] to create a connection between the robot and a web server by using 'Rosbridge' protocol, which sends the data in JavaScript Object Notation (JSON) format. According to [7] the video streaming is most heavy bandwidth-wise. ROS provides three kinds of pipeline image streaming: raw images that use the image message from ROS, Theora streams that use the video codec of the Theora, and compressed images that could use both JPEG image or PNG image compression [7].

ROS is mainly used in the local area network (LAN). As a middleware, it uses a central system for holding the path of peer-to-peer handshaking and universally related information such as parameters, robot joint state, etc. Therefore, the usual ROS operation has problems in the case where there is a need for offloading tasks to the cloud via a wide area with limited communication bandwidth [8].

More widely, the area of cloud robotics research focuses on systems that depend on any form of process that needs to be offloaded via a network to support the robot operation [6, 17]. From this definition, many cloud robotics architectures have been designed to offload as much process as possible on a remote compute node unlimited by usual resource bounds. A typical example would be the RoboEarth project that uses the cloud engine called Rapyuta [8] which is part of ROS package that is used to run processes in the cloud, utilizing a Rosbridge protocol of JSON ROS topics.

Rapyuta is a project that is customized to securely manage the network for cloud robotic environments [14]. The main use is to connect ROS with RoboEeath. Although, it supports offloading the heavy tasks to the cloud, the control in real time is still challenging, due to the network latency.

The communication network that could support the mobile robot with the real-time connection depends on the bandwidth limitation for the uplink provided by the Wireless Technology. In case of Wi-Fi 802.11 G the uplink is capped approximately at 6 bit/s [9], although that is far less than advertised speed. The uplink bandwidth in the case of applying the robot cloud is more important than the downlink, due to the volume of the data needed to be offloaded via the network.

3 SPECIFICATIONS OF HUMANOID ROBOT (NAO)

Since 2008, the humanoid robot NAO have been marketed, designed with the intent to look approachable. With an approximate weight of 4.5 kg and a height above half a meter, this robot has a human appearance. It has 11 degrees of freedom for its lower parts and an extra 14 degrees of freedom in its upper body. Regarding the hardware, the NAO

robot is fitted with a special set of sensors (e.g., ultrasound and touch) actuators for the joints, LEDs and motors. Communication is open using Wi-Fi connection and the Ethernet network. Also, it has a CPU GEODE 500 MHz

In terms of software, NAO is supplied with the NAOQi 2.0 framework that runs on Linux [10]. The framework provides an access to all the sensors of the robot, transferring the commands to actuators, reading sensors, handling Wi-Fi connections. NaoQi can execute functions in parallel, sequentially, or driven by events. NaoQi's functions can be called in C++, in Python and even in Urbi.
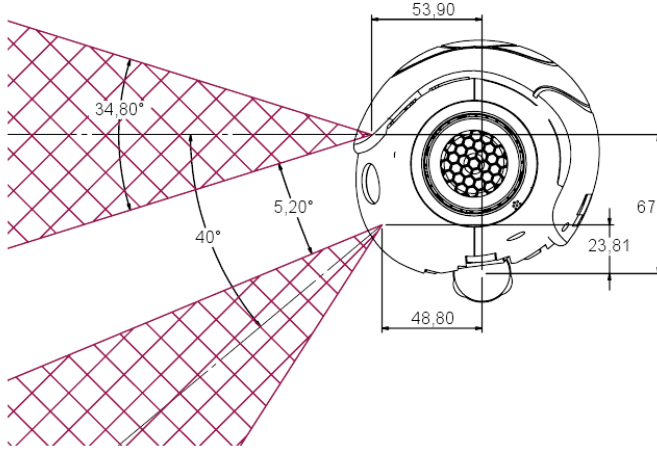
Figure 1: Field of view of NAO's video camera [11].

In this research, the video streaming will use the built-in top camera that is located on the NAO's head. The top camera provides a list of video qualities with their resolutions as shown in Table I, with a maximum frame rate up to 30 frames per second. The top camera is used for recording and video streaming during the experiment. The bottom camera was disabled during the experiments to get the result of a single camera only.

The vision system for the robot permits the robot to capture photos and streaming video. That allows the robot to perform tasks such as face and object recognition.

Table 1: Video Quality Provided by NAO Robot

| Video Quality	Resolution
K4VGA	1280x960
KVGA	640x480
KQVGA	320x240
KQQVGA	160x120

4 THE INTERACTION PROTOCOL

4.1 Computing Environments

The main computing software used in this study is the Choregraphe [10], which is developed by Aldebaran Robotics in order to control and program the NAO robot. It has embedded behaviors software modules that offer a feature to control the NAO robot, which permits the NAO to perform such complex tasks as face detection and face recognition. Furthermore, it allows the developer to access the data sensor or control actuator by just drag-and-drop behavior boxes or programming new behaviors as shown in fig. 4, which illustrate the process of face detection and face recognition. Choregraphe runs on a PC, providing users with access to a graphical user interface GUI, which utilizes all the functions delivered by NaoQi.

4.2 Communication Protocols

The experiment network architectures shown in fig. 2 demonstrate the connection between the robot and the PC that runs Choregraphe and monitor software over a local wireless-G network with a bandwidth of 54 Mbit/s in Wi-Fi connection and CAT 5 Ethernet cable with a bandwidth of 100 Mbit/s in the Ethernet connection. In both cases, the networks were free of other traffic.
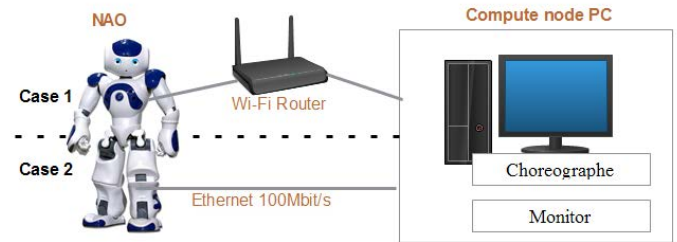
Figure 2: The network architecture for the two used cases

4.3 Experimental Set-up

In the experimental set-up, two types of software were used, Monitor 2.1.4 and Choeregraphe. In the first experiment, Monitor 2.1.4 was used to control and change the video quality and number of frames of the received video, and to change between color, and black and white modes.

The Choregraphe controls the robot behaviors in timeline action as shown in fig. 4, which illustrates the sequence of linked behaviors boxes, which is simply software to control the robot programmed with Python.

The face recognition behaviors have several components, starting with face tracking, basic awareness, then face detecting, and finally face recognition box that is responsible of run the face recognition algorithm to examine the detected face. In order to get a reliable result, all the tests of the face recognition have been done under the same circumstances,

light in the room and the distance between the face and the robot camera. These conditions could affect the face recognition result. The experiments were conducted with a physical NAO robot. The compute node is a standard PC.

The two network monitoring tools used during the tests to measure the average value of the upload link for the NAO robot were both the Tomato v.1.28 web based dashboard installed in the Wi-Fi router, and NLOAD command line tool. Both tools support a real-time bandwidth monitoring.

4.4 Face Recognition Algorithm

The Face recognition algorithm that is supported by Choregraphe is the incremental Principal Component Analysis (PCA) Algorithm [11] which is responsible for analyzing the detected face to extract the face features. The algorithm runs a Python code that is used to compare captured data with the faces saved into data sets. Fig. 3 illustrates the flowchart of the required stages for the Face Recognition process.
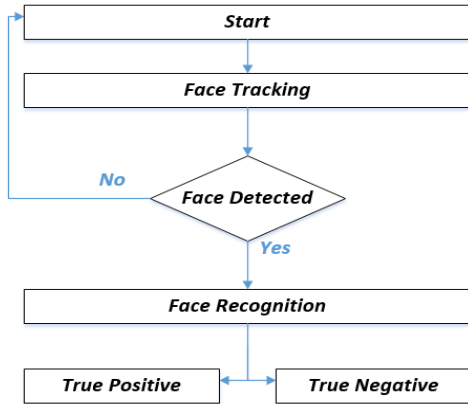
Figure 3: Flowchart of the Face Recognition process stages that are needed in NAO robot with Choeregraphe to perform Face Detection and Recognition
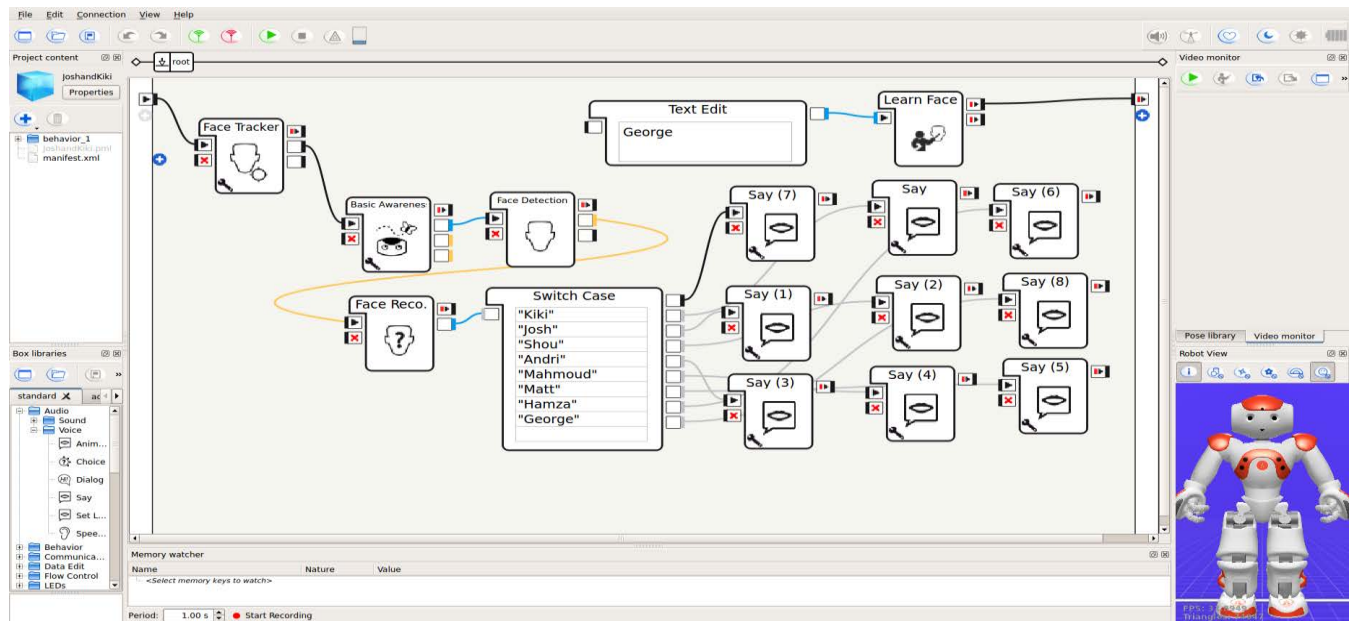
5 EXPERIMENTS AND RESULT

5.1 Local Network

In this section, a comparison of different transmission technologies such as Wi-Fi and Ethernet will be examined. In addition, four video quality streaming (KQQVGA, KQVGA, KVGA and K4VGA) will be discussed in terms of bandwidth consumption for a specific number of frames that has been sent from the NAO robot to the PC unit.

The major difference between black and white video quality and color video quality will be explained. Moreover, the face recognition accuracy for each video streaming quality and the robot learning algorithm will be addressed. Finally, there will be a discussion about the best face recognition case among all the observed tests. The bandwidth for the color video is approximately double the bandwidth for the black and white mode for a limited number of frames. For instance, by using the KQQVGA quality video stream, the bandwidth of the color video remains doubles the bandwidth of the black and white for the first 26 frames. As shown in figure 6(a) the bandwidth of 1000Kbits/s is used for only seven frames of KQVGA color video quality.

Case 1: Wi-Fi Video Streaming. In this case study, NAO robot is connected to a Wi-Fi network to examine its performance for different video streaming qualities. Fig. 5 (a) shows the bandwidth in Kbits per second for transmission of color video stream using Wi-Fi technology, whilst Fig. 5 (b) presents the bandwidth for black and white video quality. In both cases the maximum bandwidth using four different video qualities is equal to 8283 Kbit/s. This result indicates that the video stream is limited in Wi-Fi transmission technology. The error Bar for all experimental results are shown in Fig. 5 (a) and (b).

Case 2: Ethernet Video Streaming. The Ethernet network

Figure 4: Sequential behaviors that needed to perform Face recognition process in Choregraphe software

connection is used to facilitate the video streaming between the Robot and the graphical user interface (GUI), on the PC unit. This case study indicates that the limitation of the bandwidth for the video stream demonstrated in the Wi-Fi case study can be increased using an Ethernet type of transmission to a maximum data rate of 81494 Kbit/s. As shown in Fig. 6 (a), a color video stream is used with four different video qualities. KQQVGA and KQVGA required the highest data rates. After a specific number of frames the data rate will remain constant. This can be shown in the KQQVGA at the 16th frame. However, for both KVGA and K4VGA video qualities the data rate will keep increasing while the number of frames is increasing; this is true for both color video stream and the black and white video stream which is demonstrated in Fig. 6 (b). The error Bar for all experimental results are shown in Fig. 6 (a) and (b).

The main difference between Case 1 and Case 2 experiments is that the Wi-Fi data streaming is limited to up to 8283 Kbit/s. However, it is approximately equal to 82000 Kbit/s using Ethernet. Moreover, black and white video streaming has a lower data rate than the color video quality, as evident from graphs in both Fig. 5 and 6.

Case 3: Face Recognition Accuracy using Wi-Fi Video Transmission. In this case, study, accuracy of face recognition by NAO robot has been examined using four different video qualities.
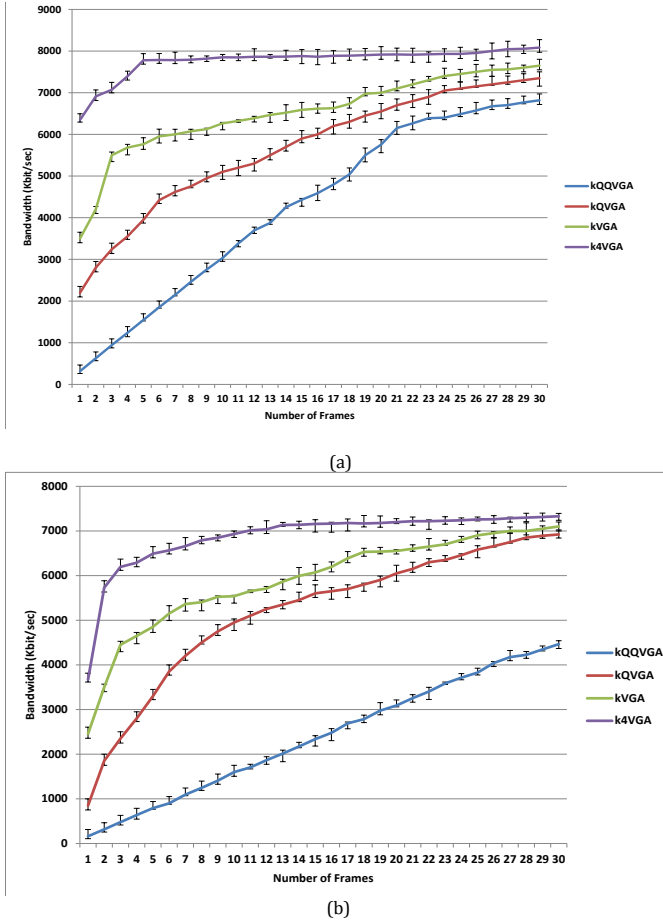
This study investigated the impact of video quality and streaming (measured in frames per second), on the accuracy of face recognition.

The face recognition model, designed in Choeregraphe, is described in Fig. 4. It is applied to eight different known faces, previously captured are stored. The captured data and person names are added to the model for each human face. The tests were designed to detect the face of a person using dynamic video streaming. The process used for face detection and recognition is shown in Fig. 3.

Initially, the robot will track any face in the surrounding area. If a face is detected the face recognition technique will start working to identify the person and find a match amongst any of saved faces. It will then start again to track any possible face movement.

The face recognition accuracy is calculated using the machine learning formula described in (1). According to [12], the minimum accepted rate for the face recognition accuracy is equal to 50%.

In this case study, the number of repetitions taken for each video quality streaming is 100, and the positive and negative detections of a human face in each experiment are added to the face recognition formula (1). We use a behavior box as counter to count the positive and negative detections.

Fig. 7 describes the results for each quality streaming, including a specific number of frames, which is sent from the
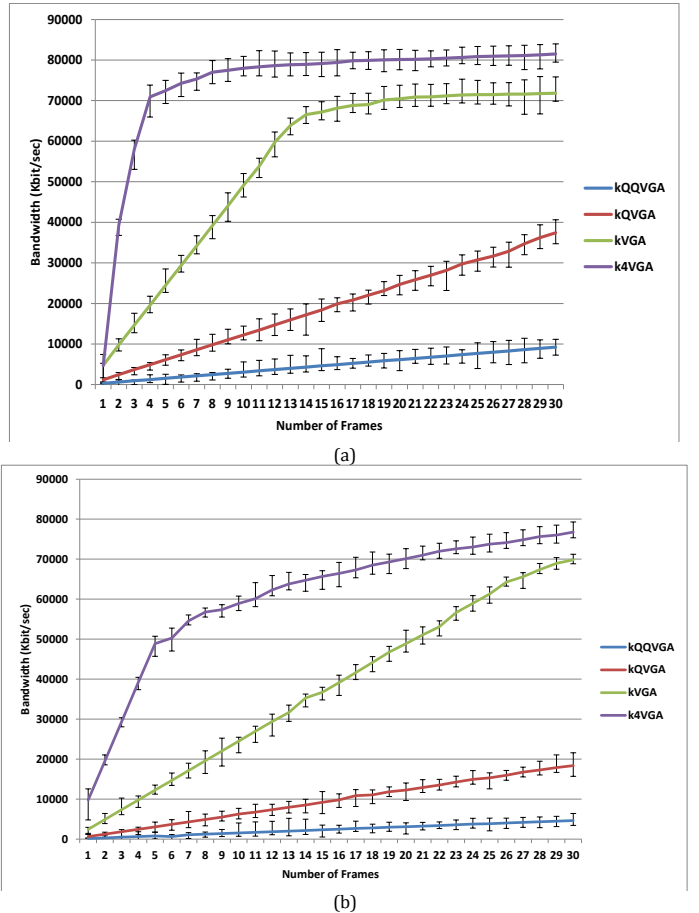

(a)

(b)

Figure 5: Wi-Fi video streaming (a) Color, (b) Black and White

(a)

(b)

Figure 6: Ethernet video streaming (a) Color, (b) Black and White

NAO robot to the PC unit interface. It is shown in Fig. 7 (a) that the maximum accuracy rate in recognizing the human face is equal to 96% using KVGA video quality and 5 fps. However, the minimum accuracy rate is equal to 56% using KQQVGA video quality and 1 fps.

The Video visualization of various video qualities can be seen in Fig. 8. It is showing four different video qualities with the default frame rate of 5 fps used in this work.

Furthermore, face detecting accuracy of the black and white video streaming, shown in Fig. 7 (b) still has a high rate of detection accuracy. The maximum detection accuracy rate is equal to 92% using K4VGA and 46% is the minimum detection rate using KQQVGA. The later is the only result that comes with the value just under the minimum accepted accuracy rate.

Several frames have a major effect on the total accuracy rate of the face recognition. As shown in Fig. 7, when the number of frames increases, the accuracy rate for KQQVGA video quality increases. This is due to the increase of the consuming bandwidth and low buffering rate at the robot side. However, for K4VGA the accuracy rate for the face recognition will decrease as the number of frames increases. It can be concluded that the consumption of the bandwidth for this video quality and the buffering rate is very high.

The face detection accuracy for KQVGA and KVGA will increase only for the first 10 frames. Afterwards, the detecting rate will keep decreasing or remains at the same percentage level.

From the results obtained in this case study an average detection rate was calculated for different video qualities shown in Figure 8. Table II shows that the optimum face recognition rate obtained from the experimental data is 74.86% for KVGA color video quality.

Table 2: AVERAGE Detection Rate

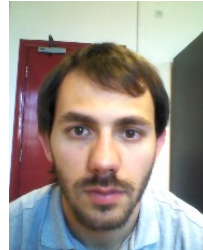| Average Detection Rate (%)	Video Quality			
	KQQVGA	*KQVGA*	*KVGA*	*KVGA*
Colour Video Quality	66.29%	74.86%	86.86 %	86.81 %
Black and White Video Quality	59.43%	69.71%	83.43 %	82.86 %
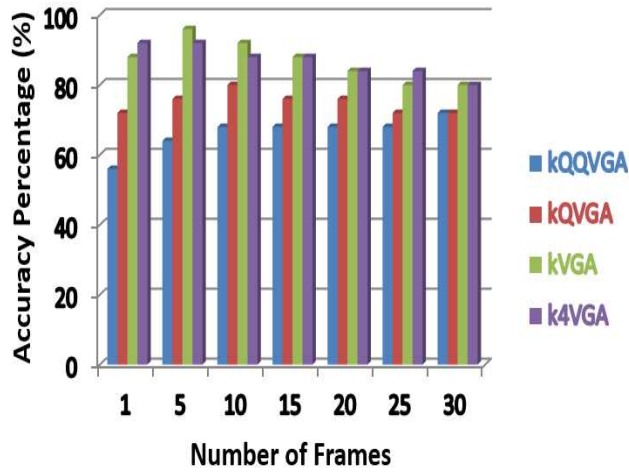
(a)
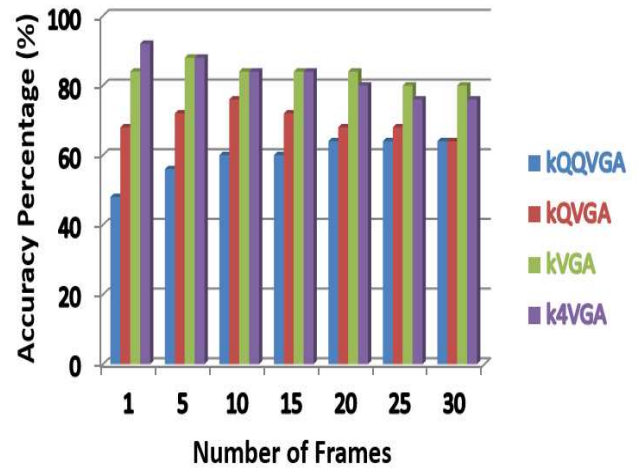
(b)

Figure 7: Face recognition accuracy rate for different video quality (a) Color video streaming, (b) Black and white video streaming
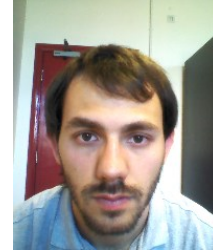

(a) KQQVGA   (b) KQVGA   (c) KVGA   (d) K4VGA

Figure 8: Video visualization of various video qualities (a) KQQVGA, (b) KQVGA (c) KVGA (d) K4VGA
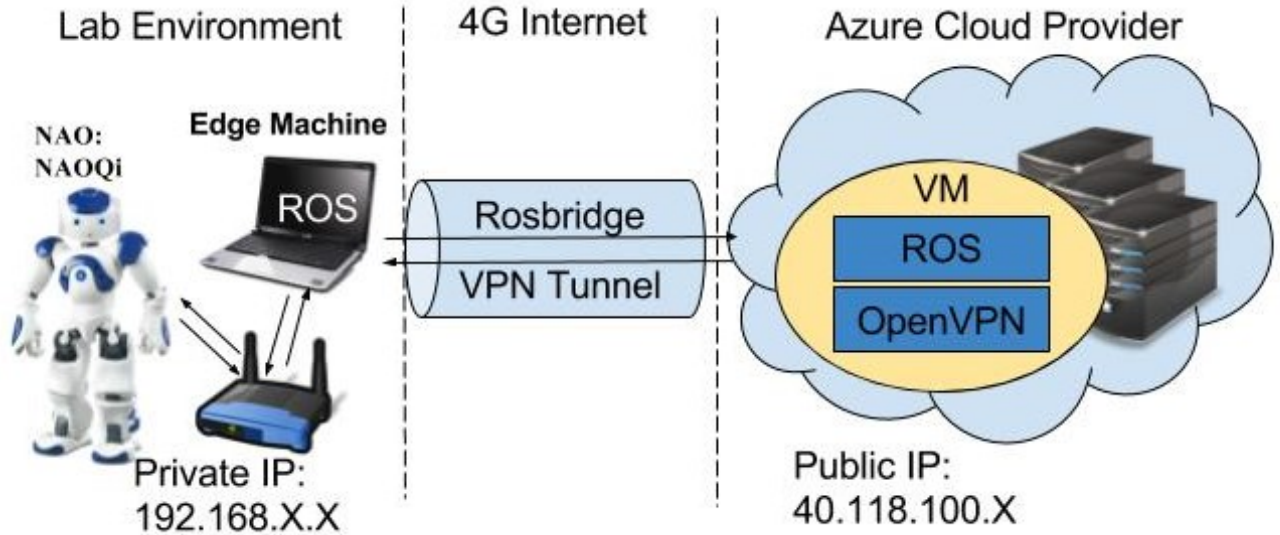
Figure 9:. The Cloud Robotic Architecture use for the experiments

5.2 Cloud robotics experiments

In this section, we focus on our cloud robotic network architecture, and the experimental process of connecting NAO to the cloud.

After analyzing the performance of the network between the robot and the compute node locally, the experiment was devised to use the cloud robotic to offload the NAO robot's sensor and actuator data to the cloud.

Fig. 9 shows the new cloud robotic architecture approach that used to link the NAO robot with the Azure cloud the experiment.

5.2.1 Network Architecture

The proposed cloud robotic system consists of several hardware and software components as shown in Fig. 9. Virtual Private Network - VPN is used to facilitate the connection between NAO robot, which operates on a private network, and a VM, which runs on the public cloud.

The communication is established though a secure VPN tunnel. The edge machine (laptop) uses a 4G dongle as a mobile gateway to the internet.

5.2.2 Experiment Set-up

The cloud site runs a VM virtual machine that has 2 cores, 4 GByte of RAM and 80 GByte of hard drive storage. The cloud provides the instance with a public IP address. The selected OS is Ubuntu 16.04 server.

The connection between the NAO robot and the cloud has been established via VPN tunnel between our lab and a VM, which runs on Azure cloud. In the lab, the robot is connected to the edge system via a Wi-Fi network. The ROS packages were installed in the cloud as a ROS master, and in the edge machine, which runs ROS as a node.

Rosbridge was used to manage the exchange of messages between ROS nodes. The use of the VPN tunnel has some communication cost, where it consumes on average 14.6 Kbit/s on the uplink and 13.4 Kbit/s on the downlink.

5.3. Overall outcomes

Based on the outcome of the experiment in the local network, the use of highest quality video over the network will not produce an optimal result, because of the lack of the bandwidth and the delay of transmission.

Therefore, in the cloud robotics experiment, we selected the KQVGA black and white video mode. This mode will be the most effective video quality to be used when offloading video data to the cloud, and will enable image processing with fair accuracy rate.

The study proves that there is a direct correlation between the quality of data and the consumed bandwidth. Moreover, the delay or buffering on transmitting high quality video streaming has an inverse relationship with the accuracy rate.

In enabled us to:

- Find the most optimized video quality to be used over the network by NAO robot when performing face recognition task in the cloud.
- Establish a secure cloud robotic network that allows the robot to use ROS on the cloud.

Therefore, the cloud has full control and access to the robot topics (sensor and actuator). In addition, these topics will be available to be used in any web-based application used to provide a service for the robots.

6 CONCLUSION

The investigation conducted in this study focused on profiling the Wi-Fi and Ethernet network performance in order to ascertain the utilization of the uplink bandwidth in the communication between the NAO robot and a computer. Three case studies were considered; Cases 1 and 2 investigated Wi-Fi and Ethernet link between the robot and the computer, and Case 3 evaluated face recognition and detection using various video qualities, in color and black and white modes.

The outcome of the experiments indicates that there is a significant impact of video streaming quality and network buffering on the face recognition process in the NAO. It is demonstrated that it is possible to obtain a real-time communication by optimizing the network bandwidth and video streaming quality over a given network. Therefore, achieve the adequate face recognition accuracy rate. The additional outcome of this study is a qualitative and quantitative measure of the video quality, type and the frame rate impact on the network bandwidth consumption on the distributed robot application.

Moreover, we established secure cloud robotic network architecture, connecting local network and public cloud – Azure via secure VPN channel that can be used with almost any robot application. An understanding of this impact could inform the development of real-time remote environment for cloud robotic or any IoT applications that aim to link the robot to an external resource, such as Web base application, utilizing the cloud resources.

Our future research work will explore further the network bandwidth requirements for offloading all the data from the NAO robot to the private cloud (Openstack) and public cloud (Azure). The overall design will utilize the proposed cloud robotics architecture and create a web dashboard to manipulate and control the robot remotely.

REFERENCES

[1] Fong, T. et al. 2003. A survey of socially interactive robots. Robotics and Autonomous Systems. 42, 3-4 (Oct. 2003), 143–166.

[2] Souissi, S. et al. 2009. Using eventually consistent compasses to gather memory-less mobile robots with limited visibility. ACM Transactions on Autonomous and Adaptive Systems. 4, 1 (Jan. 2009), 1–27.

[3] Sato, M. et al. 2011. The Ubiquitous Network Robot Platform: Common platform for continuous daily robotic services. 2011 IEEE/SICE International Symposium on System Integration (SII). (2011).

[4] Tenorth, M. et al. 2012. The RoboEarth language: Representing and exchanging knowledge about actions, objects, and environments. 2012 IEEE International Conference on Robotics and Automation. (2012).

[5] Prabha, S.S. et al. 2014. Smart cloud robot using raspberry Pi. 2014 International Conference on Recent Trends in Information Technology. (2014).

[6] Kehoe, B. et al. 2015. A Survey of Research on Cloud Robotics and Automation. IEEE Transactions on Automation Science and Engineering. 12, 2 (2015), 398–409.

[7] Toris, R. et al. 2015. Robot Web Tools: Efficient messaging for cloud robotics. 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). (2015).

[8] Hunziker, D. et al. 2013. Rapyuta: The RoboEarth Cloud Engine. 2013 IEEE International Conference on Robotics and Automation. (2013).

[9] Gass, R. and Diot, C. 2010. An Experimental Performance Comparison of 3G and Wi-Fi. Passive and Active Measurement Lecture Notes in Computer Science. (2010), 71–80.

[10] Pot, E. et al. 2009. Choregraphe: a graphical tool for humanoid robot programming. RO-MAN 2009 - The 18th IEEE International Symposium on Robot and Human Interactive Communication. (2009).

[11] Ismail, L. et al. 2011. Face detection technique of Humanoid Robot NAO for application in robotic assistive therapy. 2011 IEEE International Conference on Control System, Computing and Engineering. (2011).

[12] Cheney, J. et al. 2015. Unconstrained face detection: State of the art baseline and challenges. 2015 International Conference on Biometrics (ICB). (2015).

[13] Chang, K.-D. et al. 2011. Internet of Things and Cloud Computing for Future Internet. Communications in Computer and Information Science Security-Enriched Urban Computing and Smart Grid. (2011), 1–10.

[14] Mohanarajah, G. et al. 2015. Rapyuta: A Cloud Robotics Platform. *IEEE Transactions on Automation Science and Engineering*. 12, 2 (2015), 481-493.

[15] Copeland, M. et al. 2015. Microsoft Azure and Cloud Computing. Microsoft Azure. (2015), 3–26.

[16] Gouda, W. and Gomaa, W. 2014. Complex Motion Planning for NAO Humanoid Robot. Proceedings of the 11th International Conference on Informatics in Control, Automation and Robotics. (2014).

[17] Arunajyothi, G. 2016. A Study On Cloud Robotics Architecture, Challenges and Applications. International Journal Of Engineering And Computer Science. (Nov. 2016).

[18] Rodríguez, I. et al. 2016. NAO Robot as Rehabilitation Assistant in a Kinect Controlled System. Converging Clinical and Engineering Research on Neurorehabilitation II Biosystems & Biorobotics. (2016), 419–423.