IoT Based Image Processing Filters

Noor Abbas Khudair[1], Mohamed Ibrahim Shujaa[1], Entithaar Mhwes Zghair[1]

[1]*Middle Technical University (MTU), Electrical Engineering Technical College, Baghdad, Iraq*

Corresponding Author: Noor Abbas Khudair
Email: Noorabbas.klmn2@gmail.com

Article Info	Abstract
	Internet of things (IoT) becomes the backbone of the advanced countries and it has a real contribute to exchange the traditional style or way of practical life, even personal life into smart style, with (IoT) technology the life become more and more easy and professional. internet of things achieves various applications coordinate with sensors and standard protocols to apply what is called machine -to- machine connection (M2M), in this paper we will talk more about the concept of (M2M), the main component of internet of things and finally the common protocols that is used in network, in addition to that this work present an IOT operation with processing system using camera for capturing image and Xilinx system generator(XSG)models for designing image processing algorithms and the result of the processing is an image with black and white for edge detection and Thresholding models and gray color image for gray enhancement model.

Introduction

The significance of technology in human life is becoming increasingly important in today's world. IOT (internet of things), a recent buzzword among technologists, refers to the internetworking of things, in which diverse devices connect with one another to fulfill various purposes. Sensor layers, data center layers, and service oriented layers are all part of the IoT architecture (Dave et al., 2015). Image processing provides a variety of camera-based sensors, and the processing of the data generated by these sensors could lead to a variety of IoT-ready applications. In order to improve throughput and efficiency, image processing systems often use a distributed architecture with dedicated processing engines (Singha & Singhc, nd). The main discussed topics in this work are; (1) The concept of IoT; (2) The main component of IoT; (3) The main protocols of IoT; (4) filtering design using system generator Simulink.

The concept of IoT

At recent days we heard a lot about IoT (The internet of things) the term which will change the future of the world. The main reasons that contribute to reach the internet of things is the high speed of internet 5G, edge computing and the artificial intelligence (Porkodi & Bhuvaneswari, 2014). Those three items are the response of the foundation the internet of things. Let us speak more about them; (1) 5G: is a new technology for transmitting data with a high speed and stability transmitting through the 5G network; (2) Edge computing: is a part of cloud computing in which all the devices connected to the internet transmit their data to the main server and process this data and remove the use less information to enable the concerned place to take the decision that provide the service for all the devices connected to

the network; (3) Artificial intelligence: is a collection of algorithms that can provide data with high quality or accuracy (Abdul-Qawy et al., 2015).

To here IoT network is ready to receive information but not from the human it is from M2M Which means machine to machine to present different services to human or people (Kumar & Smys, 2018). IoT is the new name of machine to machine or M2M technology that give the permission to connect devices with others automatically using wireless network (Gubbi et al., 2013).

The main component of IoT

The IoT requires vital technological components to be brought together to build the IoT world in order to be realized as a fully integrated future Internet. We'll talk about it in this part. Enabling technologies are essential. The goal is to create a quick description of each piece and its significance in the IoT paradigm. Figure (1) bellow explains the main IoT component
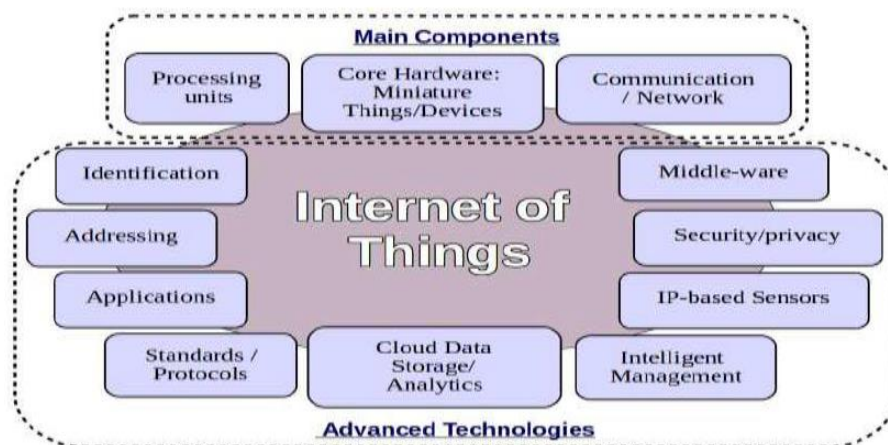

Figure 1. The main element of IoT (Kumar & Smys, 2018)

Identification and Addressing

IoT contain a large number of devices so the identification, addressing, discovering is the basic technologies of IoT. every device connected to IoT network must be individually selected to recognize each element, it's location, function and the remote connectivity across the internet. There are three technologies to detect objectives remotely (Abdul-Qawy et al., 2015).

RFID.

IPV4.

IPV6.

RFID (Radio Frequency Identification) is a wireless technology for remotely identification. which can be used for surveillance issues and following in specific time and in automatic tolls applications (Sakamura, 2006)

IPV6 (Internet Protocol Version 6) is a very important technology and it developed to address the IPV4(Internet Protocol Version 4) limitation about the newly space requirements, so IPV6 is widely used home application (Welbourne et al., 2009).

Embedded Sensors

In general sensors have different types and functions and working with IoT system required using different sensors to facilitate work in difficult conditions (Jara et al., 2013), intelligent sensor.

Protocols And Middleware

As we mentioned above IoT system contain large number of devices with different capabilities which need for exchanging data and information so this device must connect to each other in a way or another, this way is called protocols (standard protocols), or in short it is a set of rules the suitable language between devices in IoT (Atzori et al., 2010).

The middleware software layer is the basic in the huge network that have different devices and functions that facilitate the interaction between "internet" and "things".

Cloud Based Storage and Analytics

A lot of IoT applications require a huge data or information and there for huge storage, huge processing, high speed to flow data and so that we need for smart algorithms to reach to the useful information and provide them to the end user (Lee & Lee, 2015).

Applications

IoT application can be divided into three parts according to the job or the function such as data collection, analytics, real time decision. Abeillé (2008) states IoT has wide application in various area of life, the figure below obtains IoT application as shown in figure 2:
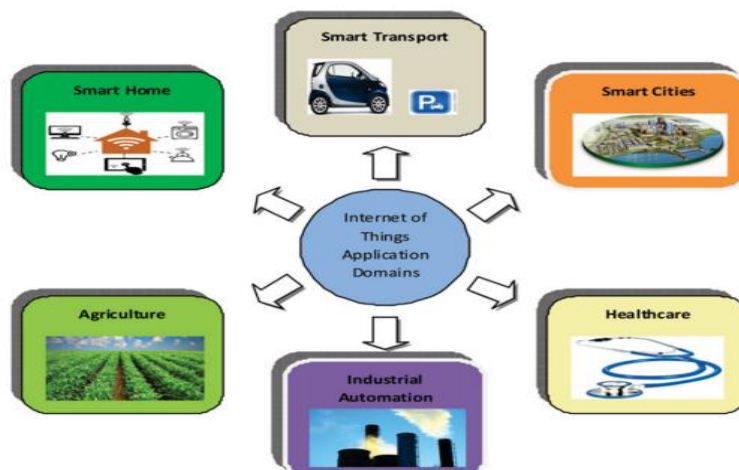

Figure 2. The main application of IoT. (Abeillé et al., 2008)

The aim of entering M2M in our daily life applications is to make life more easy and enhanced the goodness of human live.

Core hardware

As we mentioned above that IoT contain a large number of devices, these devices must be in special component such as memory, processing unit, power supply and so on to help the M2M connection between different devices and to enable sharing information in intelligent system (Gupta & Quamara, 2020).

The main protocols of IoT

Because of the large scale network of IoT and contain it the constraint and smart devices, the matter that make protocol is very important to enable an efficient communication for various applications and services (Gupta & Quamara, 2020), let's take quick look at these protocols:

MQTT (Message Queue Telemetry Transport)

Is a standard protocol for the internet of things, it is publisher /subscriber messaging protocol, it is easy to implement, runs on TCP/IP protocol, MQTT feature is lightweight protocol provide one –to –many message transmitting, low bandwidth, used for exchange data between constraint device and server application [16] MQTT architecture is shown in figure 3.
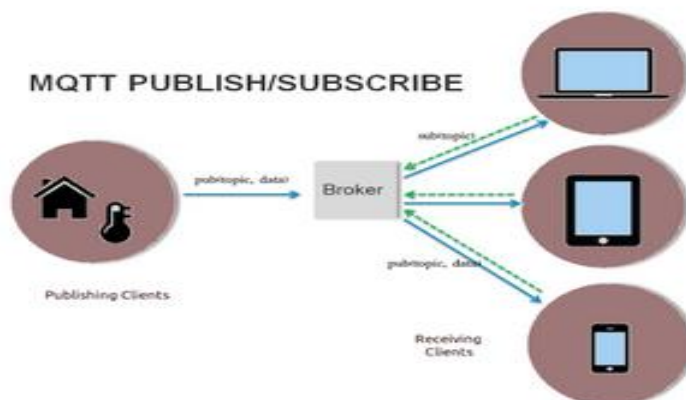

Figure 3. MQTT architecture (Tsai et al., 2014).

COAP (constraint application protocol)

It is request /response protocol specially use for web data transmitting in application layer it designed to enable constraint devices to connect to each other and it designed also for machine to machine application. COAP features are low overhead, synchronous message exchange, low power, runs over UDP not TCP Protocol, asynchronous message exchange so the application that is suitable is smart city and building automation (Bhawiyuga et al., 2017). see figure 4.
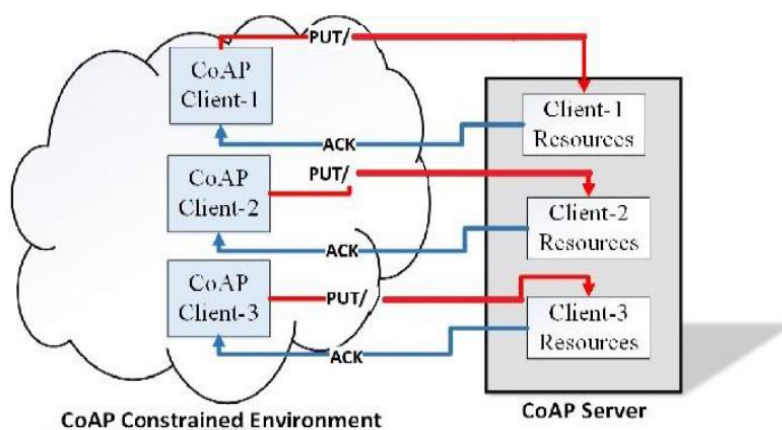

Figure 4. structure of COAP Protocol (Bhawiyuga et al., 2017)

Hyper Text Transfer Protocol (HTTP)

It's a web-based data distribution protocol that operates at the application layer, but it doesn't meet the requirements of the Internet of Things. A comparison was done with the CoAp

4

protocol, and it was discovered that the CoAp approach saves more energy (Villaverde et al., 2012).

Filtering Design Using System Generator Simulink

The Xilinx System Generator is a design, analysis, and targeting tool. It is a design process for developing FPGA-compatible DSP architectures. It combines the MATLAB and Xilinx ISE software suites into one. XSG uses the following design flow: Add the system generator icon to an executable Simulink design. Using Xilinx Core Generator, the system generator will create the RTL automatically, the system generator will build the RTL and test bench for the provided Simulink design using the Xilinx Core Generator. It will generate the ucf file automatically, which will aid in the testing of the algorithm (Han, 2015). The System Generator block specifies the type of FPGA board to be utilized, as well as a number of other parameters such as clock speed, compilation type, and analysis.

Related Work

Joshi et al (2005) provided a configurable system for removing noise from image using statistical modeling of wavelet coefficient.

ChapIe et al (2014) produced solution for real time image processing that required handling for the huge data of pixels by using pipelining device like FPGA for reducing the execution time for processing algorithm, in addition to that this work used Sobel edge detection algorithm to detect pixel in gray scale image Xilinx ISE Design Suite-14 software platform.

Doshi et al (2015) described the implementation of cloud using Xilinx system generator that can be used distantly, the cloud approach can be reached using Python and Hypertext Preprocessor (PHP) for running Xilinx environment on the system server.

Venkatesh H et al (2015) Introduced an IoT scenario to odd the sign of traffic light by transmitting a message and hence ambulance can reach the hospital without any lateness and without loss time, they utilized an embedded system IoT to develop a model for the ambulance coming in the path. GSM cellular network and GPS system used for IoT technologies.

S. Sicari, et al., (2015) provided heterogeneous technologies of IoT which provide different services in various application framework, discussed the requirements of security and privacy as a fundamental rule in IoT network, scalability issues and flexibility infrastructure is used to deal with menace dynamic media. this work proposed the challenges field of IoT security and suggested some notes for future work.

YIN ChuanTao et al (2015) Presented the most important challenges faced the smart cities fundamentals and obtained how people in cities help the manufacturing field and improve production efficiently.

Huang et al (2017) suggested a new computing model to solve the problem of the huge number of devices connected to the cloud services, fog computing model provided analytic and intelligent employ to the user with fast response. In IoT application publish/ subscribe was the best model for lower Bandwidth and less command processing network. MQTT and COAP support publish /subscribe model.

Kumar et al., (2017) proposed various features in different cryptography libraries in IoT environment along with security measurement in different communication models would lead to more secure and reliable IoT infrastructure for wide adaption.

Methods

Remotely image processing is one of the most important application for detecting or extracting features from image, so the aim of this work to present the research methodology of developing and implement filtering design distantly using camera for capturing and raspberry pi microcontroller device. Then use a robust image processing algorithms based models by a software like MATLAB Simulink version R2012a and ISE design version 14.7 which presenting FPGA input as hardware but we will content with software testing for Xilinx models.

The first stage in the IoT system is capturing a photo from the camera of 5 pixels that connected to raspberry pi, the size of photo image is approximately 41 KB with dimension of 288*384*3, type JPEG.as shown in figure 5.
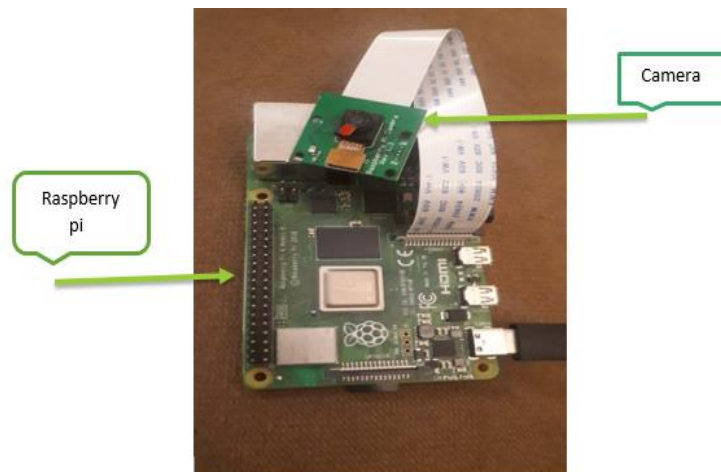

Figure 5. raspberry pi device connected with camera

The second stage in raspberry pi where the transmitting operation is beginning. We require an easy and available linking way to send data from the Raspberry Pi to a server all of the time, and the Raspberry Pi provides a link via Wi-Fi using sockets. Figure 6 explain the system.
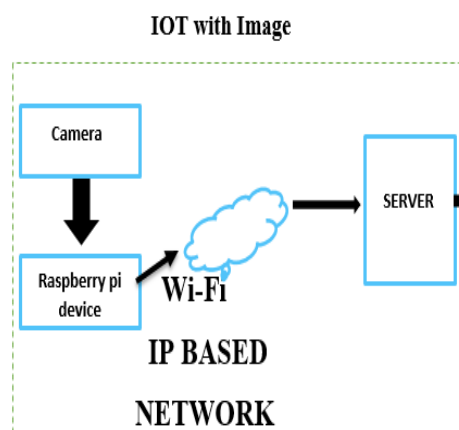

Figure 6. IoT system

Many applications are concurrently running over the web such as audio & video, file transfer and so on, in our system we have the file transfer is the capturing image and the proper communication to take place between the client (raspberry pi) and the server in laptop. The file transfer must agree with HTTP application level protocol. HTTP (hypertext transfer

6

protocol) is the most popular application layer protocol used in the internet .so the HTTP send a post request message to HTTP server, as shown in figure 7 below:

Figure 7. connection network

http protocol is runs over TCP/IP, TCP (Transport Control Protocol) is reliable protocol and works in transport layer, each packet has sequence number and if the will be retransmitted if it is not received by the receiver, so the TCP is guarantee in packet delivery.

IP (Internet Protocol) is a network layer protocol deal with the network addressing and routing where the connection network between the client and server is local network. Now when the image file is received by the server the rest of the system will be with the processing of image. The processing will be done by Simulink using Xilinx system generator (XSG).

(XSG) Read image model is done by two stages

The block diagram of the pre-processing stage is shown in Figure 8. As FPGA works with fixed point processing, the main task to be performed is picture pre-processing. To convert the matrix form to text form, the image must be translated from 2D to 1D. A frame conversion block set has been added to boost throughput. However, it is not required to use it. The data is then unbuffer as input to the gateway in block. The gateway in a block that receives transformed data and sends it to the FPGA.As shown in figure 8 :
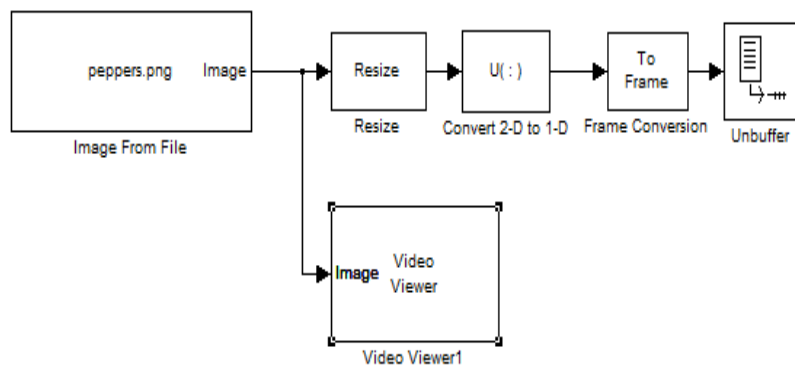

Figure 8. image preprocessing stage

The block diagram of the post-processing stage is shown in Figure 9. To display an image after completing a set of activities on a fixed point image, the result must be converted to 2D. The result undergoes data conversion to produce an uint8 image. The data is then buffered to build a text, which is subsequently transformed back to 2D and presented. As shown in figure 9:
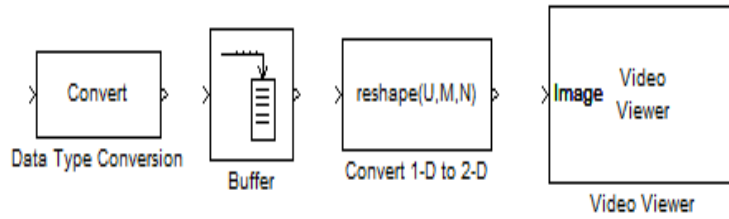
Figure 9. image post processing stage

The first processing will be with edge detection model

Filtering and masking with a proper filter mask is all that is required for edge detection. Edge, Sobel X, Sobel Y, Sobel X-Y, Blur, Smooth, Sharpen, Gaussian, or Identity filtering coefficients can be retrieved using the 5x5 Filter Mask. To avoid this problem, the system generator uses a 5x5 filter block to generate a delay for filtering operations. Figure (10) explains edge detection algorithm:
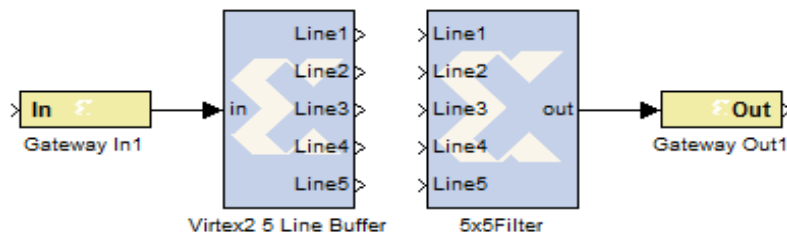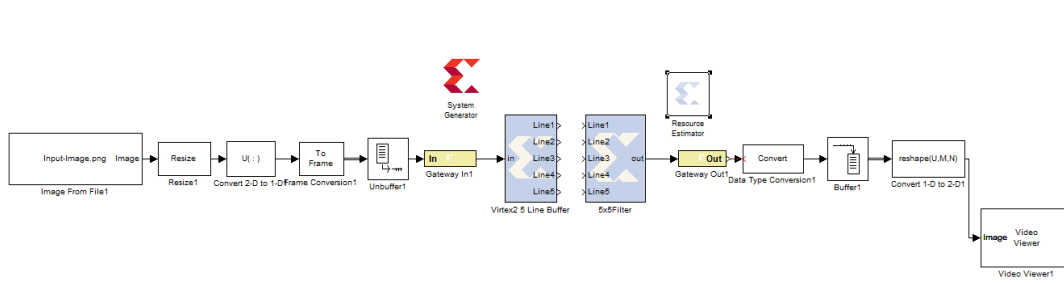

Figure10. edge detection algorithm

Figure 11. edge detection model

The second model is for gray scale enhancement

By just adding additional illumination to certain photographs, they can be improved for perception. A single dimensional image signal is resolved for grayscale images, and multidimensional R|G|B image signals are resolved similarly for color images. As shown in figure 12:
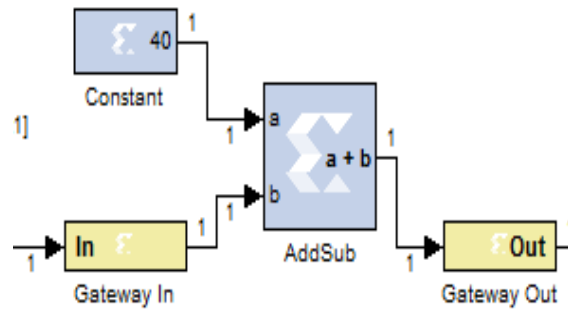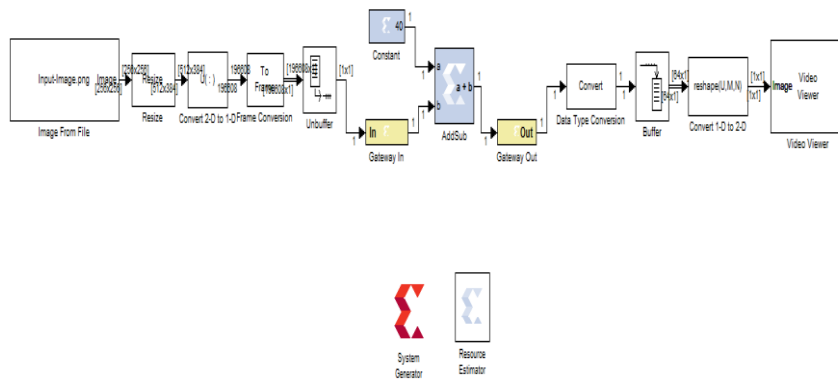
8

Figure 12. gray scale enhancement algorithm

Figure13. gray scale enhancement model

The third algorithm is Thresholding

Thresholding an image means turning all pixels above a given threshold level white while leaving the rest black. A suitable constant, such as 55, is utilized to implement the method, and a Mux is used to replace the thresholds with white. As shown in figure 14.
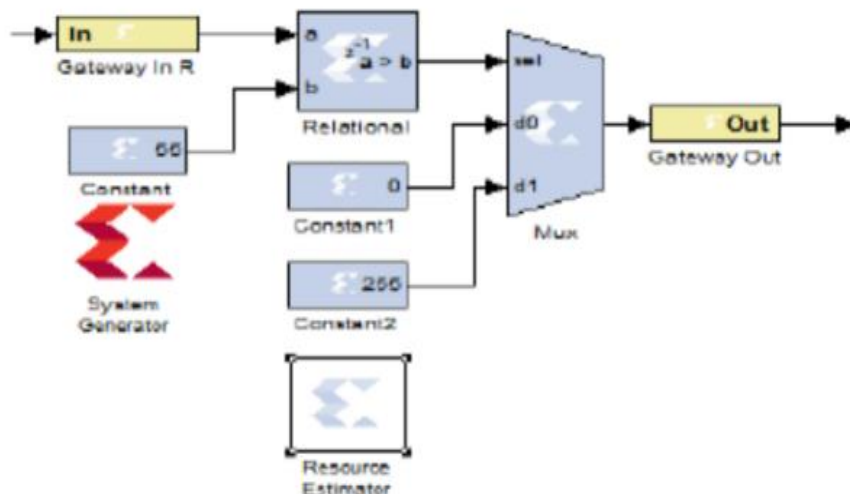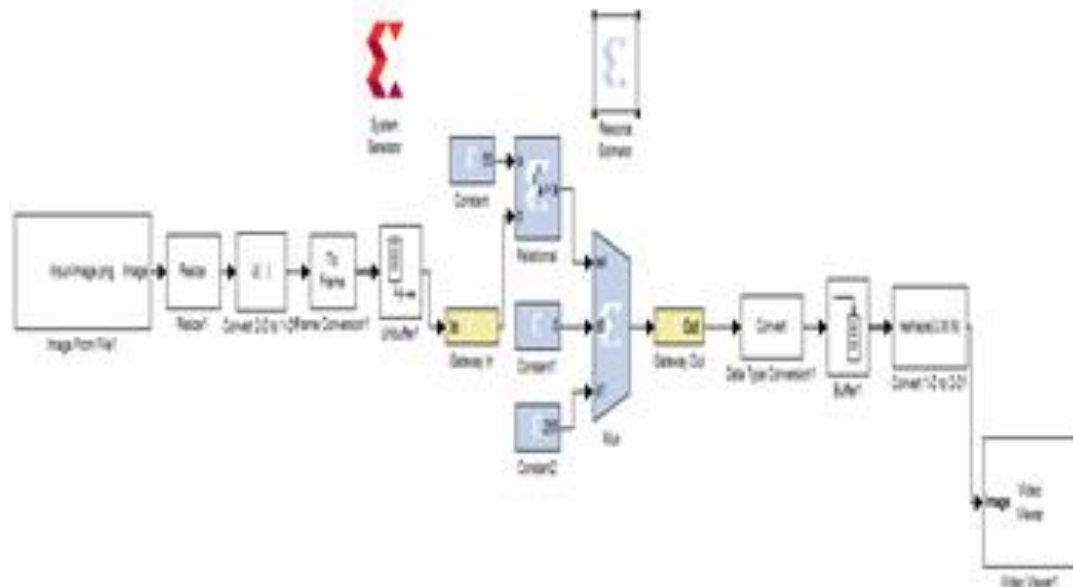

Figure14. Thresholding algorithm

Figure15. Thresholding model

Results and Discussion

Result of IoT Image file transferring and MATLAB SIMULINK

After the post request using http request message the image is received by the server with number of sequence packets that arrived successfully without

any losses, image is transmitted with RGB type and it is received in the same type. See figure 16 obtain the original image

Figure 16. The received image (original image)

The figure bellow explains the result of edge detection design:

Figure 17. edge detection model

Figure 18 obtain the result of the gray scale enhancement design:

Figure 18. gray scale enhancement model

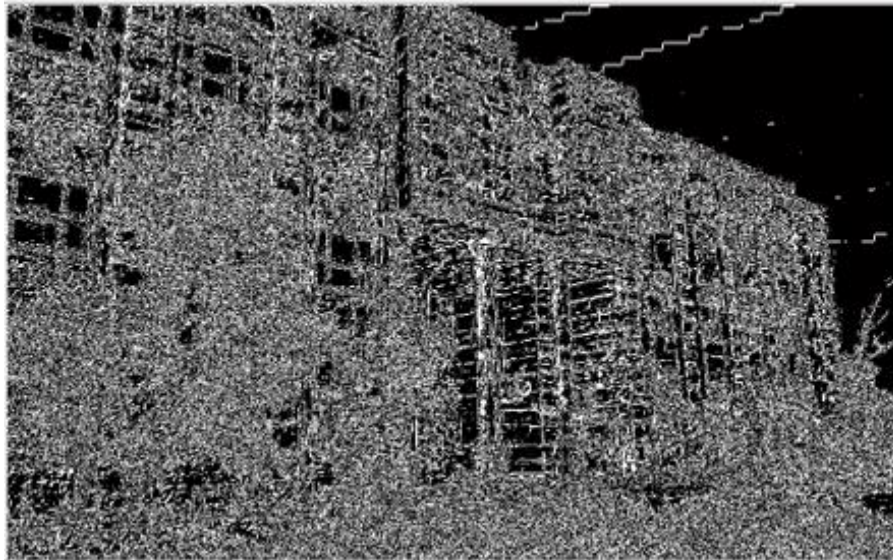The result of Thresholding model is shown in figure 19 bellow:

Figure 19. result for Thresholding model

Conclusion

Today With the importance of internet for everybody whether in private life or in practical live, we can imagine the world with out of internet specially with the smart application that presented to us, all the objectives around our environment can be connected to internet with a large network containing different sensors with standard protocol for IoT, and it provides the chance for people to control things over distance Without the need to be in a specific place to deal with a specific device, hence the IoT system that is used in this work is successfully using raspberry pi as an IoT device and HTTP post request for transferring the captured image file and server for receiving and store image. The future work will be about processing the image that stored in the server using training system and robost algorithm for processing using also hardware platform

References

Abdul-Qawy, A. S., Pramod, P. J., Magesh, E., & Srinivasulu, T. (2015). The internet of things (iot): An overview. *International Journal of Engineering Research and Applications*, *1*(5), 71-82.

Abdul-Qawy, A. S., Pramod, P. J., Magesh, E., & Srinivasulu, T. (2015). The internet of things (iot): An overview. *International Journal of Engineering Research and Applications*, *1*(5), 71-82.

Abeillé, J., Durvy, M., Hui, J., & Dawson-Haggerty, S. (2008). Lightweight IPv6 stacks for smart objects: the experience of three independent and interoperable implementations. *IPSO White Paper*, *2*.

Atzori, L., Iera, A., & Morabito, G. (2010). The internet of things: A survey. *Computer networks*, *54*(15), 2787-2805.

Bhawiyuga, A., Data, M., & Warda, A. (2017, October). Architectural design of token based authentication of MQTT protocol in constrained IoT device. In *2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA)* (pp. 1-4). IEEE.

Dave, Z., Dhote, S., Joshi, J., Tambe, A., & Gengaje, S. (2015). Network on chip based multi-function image processing system using FPGA. In *2015 International Conference on*

Advances in Computing, Communications and Informatics (ICACCI) (pp. 488-492). IEEE.

Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems*, *29*(7), 1645-1660.

Gupta, B. B., & Quamara, M. (2020). An overview of Internet of Things (IoT): Architectural aspects, challenges, and protocols. *Concurrency and Computation: Practice and Experience*, *32*(21), e4946.

Gupta, B. B., & Quamara, M. (2020). An overview of Internet of Things (IoT): Architectural aspects, challenges, and protocols. *Concurrency and Computation: Practice and Experience*, *32*(21), e4946.

Han, N. S. (2015). *Semantic service provisioning for 6LoWPAN: powering internet of things applications on Web* (Doctoral dissertation, Institut National des Télécommunications).

Jara, A. J., Ladid, L., & Gómez-Skarmeta, A. F. (2013). The Internet of Everything through IPv6: An Analysis of Challenges, Solutions and Opportunities. *J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl.*, *4*(3), 97-118.

Joshi, J., Nabar, N., & Batra, P. (2006, August). Reconfigurable implementation of wavelet based image denoising. In *2006 49th IEEE International Midwest Symposium on Circuits and Systems* (Vol. 1, pp. 475-478). IEEE.

Kumar, R. P., & Smys, S. (2018). A novel report on architecture, protocols and applications in Internet of Things (IoT). In *2018 2nd International Conference on Inventive Systems and control (ICISC)* (pp. 1156-1161). IEEE.

Lee, I., & Lee, K. (2015). The Internet of Things (IoT): Applications, investments, and challenges for enterprises. *Business Horizons*, *58*(4), 431-440.

Sakamura, K. (2006). Challenges in the age of ubiquitous computing: a case study of T-Engine, an open development platform for embedded systems. In *Proceedings of the 28th international conference on Software engineering* (pp. 713-720).

Singha, A., & Singhc, J (nd). Image Processing and IOT Based Applications. ijisrt.com
Porkodi, R., & Bhuvaneswari, V. (2014). The internet of things (IOT) applications and communication enabling technology standards: An overview. In *2014 International conference on intelligent computing applications* (pp. 324-329). IEEE.

Tsai, C. W., Lai, C. F., & Vasilakos, A. V. (2014). Future internet of things: open issues and challenges. *Wireless Networks*, *20*(8), 2201-2217.

Villaverde, B. C., Pesch, D., Alberola, R. D. P., Fedor, S., & Boubekeur, M. (2012, July). Constrained application protocol for low power embedded networks: A survey. In *2012 sixth international conference on innovative mobile and internet services in ubiquitous computing* (pp. 702-707). IEEE.

Welbourne, E., Battle, L., Cole, G., Gould, K., Rector, K., Raymer, S., ... & Borriello, G. (2009). Building the internet of things using RFID: the RFID ecosystem experience. *IEEE Internet computing*, *13*(3), 48-55