Short Paper of CSCI-ISOT : Mesh-IoT Based System For Large-Scale Environment

1st Archit Gajjar
Department of Engineering
University of Houston - Clear Lake
Houston, USA

2nd Xiaokun Yang - Contact Author
Department of Engineering
University of Houston - Clear Lake
Houston, USA
YangXia@uhcl.edu

3rd Hakduran Koc
Department of Engineering
University of Houston - Clear Lake
Houston, USA

4rd Ishaq Unwala
Department of Engineering
University of Houston - Clear Lake
Houston, USA

5th Lei Wu
Mathematics and Computer Science Department
Auburn University at Montgomery
Montgomery, USA

6th Jiang Lu
Department of Engineering
University of Houston - Clear Lake
Houston, USA

***Abstract*—This paper presents an Internet-of-Things (IoT) architecture by integrating a Synology cloud server, edge computing systems, and physical networks. More specifically, we have established a physical network combining two subsystems 1) non-standardized Bluetooth Low Energy (BLE) Mesh network, and 2) a security monitoring system. The BLE Mesh system has one IoT host connected with three BLE devices, enabling to extend the communication distance by using one or two relays. The monitoring system consists of a Passive Infrared Sensor (PIR) and a webcam with multiple solutions for recognizing a human face. Two algorithms, Low Binary Pattern Histogram (LBPH) and Deep Metric Learning (DML), have been implemented and evaluated on different benchmarks. Experimental results show that the DML-based computation can reach 99.38% accuracy with almost 400 ms latency for recognizing a single face in frames of images.**

The future work will focus on testing the cloud service by integrating a Synology D218+ server, as well as improving the computation speed of facial recognition on pure hardware design on field-programmable gate array (FPGA). The aim of our work is to provide a robust IoT-Edge-Cloud system which can be deployed on the large-scale applications and processes data much faster compared to traditional cloud computing system due to the perks of parallel computing on FPGAs at the network edge.

***Index Terms*—Edge computing, Internet-of-Things (IoT), Facial recognition**

I. Introduction

Due to the recent innovations in the field of Internet-of-Things (IoTs), there has been a tremendous amount of growth of the devices that are connected to the Internet. These devices are producing gigantic data consuming a lot of energy. Thus, it creates a difficulty in terms of deployment, management, intelligently processing the data in large-scale systems such as industrial factories, agricultural farms, or buildings [1], [2].

Under this context, there were many projects and publications available on topics of, separately, artificial intelligence, data security, and mesh network. For example, recently machine learning such as object/facial recognition has made an even preeminent impact for the surveillance systems preventing crime [3], [4]. To improve the energy-efficiency, an application-specific design of facial detection on field-programming gate array (FPGA) has been further presented [5].

Additionally, Ref. [6] and [7] have provided a simple and inexpensive way to connect together four fabricated Bluetooth-enabled boards and establish a mesh network, though the literature only provides a solution to sense the environment and actuate based on a control algorithm. Finally, when an IoT system with large amount of devices was established on wide-spread environment, security becomes one of the challenges that need to be tackled scrupulously [16]–[18].

The aforementioned prior works are stimulating source of motivation and bolstered us to propose a pioneering architecture. The main challenge is to provide an amalgam of them, capable of controlling huge installed embedded devices along with a security alert system. In this paper, therefore, we presented an IoT architecture with an integrated IoT-Mesh-Cloud system, offering an efficient solution for a combination of cloud control, edge computing, and Bluetooth Low Energy (BLE) Mesh network. The main contributions of this paper are:

- We presented a robust physical network by using BLE Mesh technology, which can be implemented on large-scale environments like agricultural farms, industrial factories, or commercial buildings. Experimental result shows that the maximum communication distance between two BLE-enabled devices is able to be extended from point-to-point 57 feet to group-to-group 77 feet by using one BLE board as a relay. Due to the BLE Mesh network's potential to be expanded on the far-reaching framework and integration with IoT security system, humongous rife data creation is expected.
- We presented face recognition with two algorithms - the Low Binary Pattern Histogram (LBPH) and Deep Metric Learning (DML), and further evaluated the performance in terms of recognition speed and accuracy with different

978-1-7281-1360-9/18/$31.00 ©2018 IEEE

DOI 10.1109/CSCI46756.2018.00197

benchmarks. As an example on the same platform, the single face recognition using DML achieves higher accuracy (99.38%) but consumes longer time (almost 400 ms) compared to applying the LBPH algorithm. However, the LBPH is able to reduce the computation latency to 10% compared with the DML operation but the accuracy is just around 50%. When testing on a Raspberry Pi 3, the computation speed of executing the LBPH algorithm would be reduced to more than 40 ms.
- We presented an integrated IoT-Mesh-Cloud architecture as a work-in-process platform. To combine the mesh network (low data rate) and the face recognition system (high data rate), a synology server will be used in the future as a data center to display and make decision based on data collected and analyzed. Moreover, the FPGA based design will be applied to improve the computation speed on high-bandwidth data processing.

The organization of this paper is as follows. Section II briefly introduces our proposed work, including cloud server, edge nodes, and deployed systems. Section III discusses the implementation of the system and Section IV shows the preliminary results. Finally, Section V presents the concluding remarks in our target architecture and Section VI discusses the future work.

II. Proposed Architecture

Fig. 1 displays the hardware architecture mainly containing three fragments - Cloud Server, Edge Nodes, and Deployed Systems.

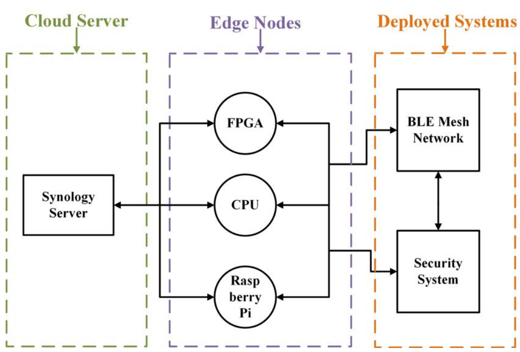


Fig. 1. The Proposed Target Architecture

A. Cloud Server

On the left side, the green box in Fig. 1 is the cloud server which is connected to all edge nodes. The cloud server is connected to the sensors and the security system through the edge nodes.

To serve the purpose of cloud computing, we have set up windows based Synology DSM v.6.2 server [10]. The utilization of server provides, with higher latency than edge nodes, a manipulation of deployed systems. It can also handle storage of formulated logs and essential data. Furthermore, any device connected to the network can access the data if needed.

B. Edge Nodes

In the middle, the purple box in Fig. 1 includes all edge nodes for the whole architecture. Basically, an edge node connects the deployed systems to cloud and helps computation time, data storage, and improves latency.

We propose three various types of edge nodes to the system such as an FPGA accelerator, CPUs, and Raspberry Pis. These devices will assist in improving the overall computation time for the task and filtering data needs that to be stored on the cloud, discard the monotonous data, and ignore garbage ones. The benefits of processing data at the network edge include low latency computation, efficient usage of the bandwidth, and security and privacy by keeping data at the local network.

C. Deployed System

On the right side, the orange box in Fig. 1 shows multiple integrated systems. In our target architecture, we have two deployed systems which are the security system and BLE Mesh system connected to the edge nodes.

The BLE Mesh network is a production of four fabricated boards using the APlix CSR1020 modules which is appropriate for low-power and limited complexity IoT applications [14]. There is one main host connected with three other boards. Each board incorporates a specific task including temperature and humidity sensor, Light-emitting diode (LED), and motor. On the other side, the security system exhibits a passive infrared sensor (PIR) sensor and a camera to detect and recognize the face(s) in the surroundings.

III. Implementation

This paper presents an ongoing project. So far, we have developed two systems: 1) BLE Mesh network and 2) Security system separately, and have integrated them to work together as deployed systems. The detail of theses systems and the alert system are presented in this section.

A. BLE Mesh Network

As described earlier, the four manufactured boards perform diverse tasks creating the whole system on the foundation on BLE protocol [15]. Using the star topology and numerous sensors, we can establish this system on giant setting [7].

Fig. 2 illustrates the star topology and how the sensors are orchestrated in the network. The maximum range between any two sensors could approximately be 57 feet. As displayed in Fig. 2, the host is only connected to LED and actuator since there are in the defined range of 57 feet which is called as group 1. The host is not capable of communicating with humidity/temperature sensor directly due to the range. When group 2 is formed with humidity/temperature sensor and actuator, the host can communicate with the sensor through one node, actuator, between them. According to test results, the measured distance between host and sensors with a node between them is maximum of 77 feet. Thus, we can deploy the whole system on large-scale environment covering the whole building, factory, or farm.

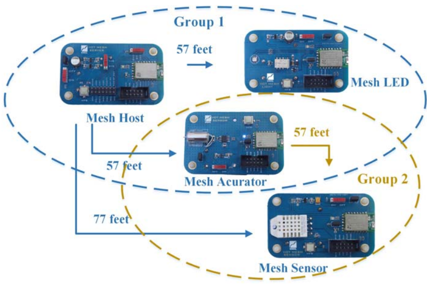


Fig. 2. Star Topology Configuration

Those boards communicate on specific self-defined commands with each other; The commands are either control or data packets facilitating actuation for LED/motor, transmitting sensors data, and receiving sensors data. An example for each task is shown in the Fig. 3(a), Fig. 3(b), and Fig. 3(c).

Header (2B)	Length (1B)	Type (1B)	On/Off (1B)	Red (1B)	Green (1B)	Blue (1B)	Check (1B)	Stop (2B)
FA, F5	07	F1	00~01	00~FF	00~FF	00~FF	XX	0D, 0A

(a) Configuration for LED

Header (2B)	Length (1B)	Type (1B)	On/Off (1B)	Check (1B)	Stop (2B)
FA, F5	04	F3	00~01	XX	0D, 0A

(b) Transmitting Sensor Packet

Header (2B)	Length (1B)	Type (1B)	Humidity (2B)	Temperature (2B)	Check (1B)	Stop (2B)
FA, F5	07	F3	0000~FFFF	0000~FFFF	XX	0D, 0A

(c) Receiving Sensor Packet

Fig. 3. BLE Mesh Board Control Packets

B. Security System

On large-scale systems, security is one of the main issues to be taken care of. We have implemented a smart security system which can detect a motion with a PIR sensor and notify the designated person if needed. Additionally, a detected motion enables the camera (Logitech C270) which tries to detect and recognize if there is a human in the captured frame or not. If a human is detected and is not in the database, then an alert notification is sent. We have also implemented two algorithms to detect and recognize the face(s): LBPH and DML.

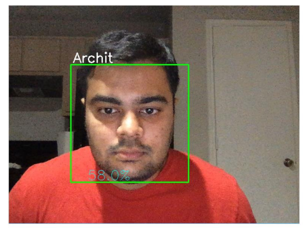
The LBPH is one of the algorithms provided by Open-CV libraries [11]. While the LBPH provides lesser accuracy compared to the DML algorithm, it has lower computation time for the same scenario. The LBPH was implemented on Raspberry Pi 3 and Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz with 16 GigaByte RAM. Due to the lower computation time, Raspberry Pi was feasible enough to implement the LBPH face recognition algorithm [8], [9].

The DML is a face recognition algorithm from the dlib C++ library [12], [13]. The algorithm provides one of the highest accuracies but also drains most of the computation power. The contrivance was successfully achieved on the same CPU -Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz with 16 GigaByte RAM.

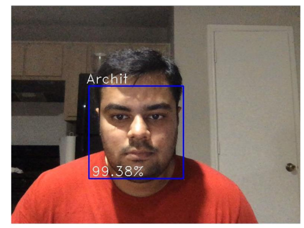
C. Alert System

Fig. 4 shows the screenshots of face recognition using LBPH and DML algorithms. Fig. 4(a) is a screenshot of recognizing person (Archit) by using LBPH. The accuracy is highly dependent upon several factors including the lighting of surrounding and the distance between human and sensor. In the LBPH, initially, we need to train the data-set after each update of the current data-set.

Fig. 4(b) is a screenshot with DML based face recognition system on CPU identifying a human. There is no need to train data-set in DML.



(a) LBPH Based Face Recognition



(b) DML Based Face Recognition

Fig. 4. Face Recognition with LBPH and DML

After the system recognizes the face(s) using Simple Mail Transfer Protocol (SMTP), the authorial person is notified providing the attachment of the image captured. In the case of a false alarm, the authorities may decide to resolve the problem by checking the attachment if they are not in the near proximity of the area.

Fig. 5 is a screenshot of a security alert from the system. Importantly, the provided screenshot is merely an example of alerting an authority. The actual system would only notify when the human face is detected and not recognized from the trained data-set (for LBPH) or database (for DML).

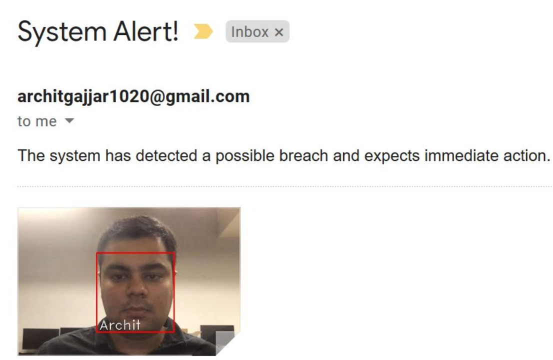


Fig. 5. Email Notification

IV. System Evaluation

In this section, the distance results for BLE Mesh technology and the performance of the security system on different platforms with diverse algorithms are discussed.

A. Mesh Network

Table I represents the maximum distance between two mesh network boards without any node between them which is 57 feet. Although any board would not be able to communicate with other boards if a distance between them is greater than 57 feet, it can be increased to 77 feet by introducing one more board between them as a node or relay.

TABLE I
DISTANCE RESULTS FOR MESH NETWORK

Maximum Distance	*No. of Node(s)*
57 feet	0
77 feet	1

B. Security System with Face Recognition

Table II displays time taken, in seconds, by different devices along with both recognizing methods. The computation time utterly relies on the hardware or benchmarks. With CPU based implementations, the face recognition with DML spends around $10\times$ of the needed for the computation on LBPH. When executing the same algorithm on Raspberry Pi, the latency is further increased by more than $10\times$.

TABLE II
FACE RECOGNITION PERFORMANCE

Devices	*Recognition Time(s)*	*No. of Face(s)*	*Method*
Raspberry Pi	0.41 - 0.45	1	LBPH
CPU	0.035 - 0.037	1	LBPH
CPU	0.391 - 0.398	1	DML

Table III shows the accuracy of recognizing a face on different systems with the same environmental setting. When training a data-set with 300 captured pictures of a human face, the LBPH provides extremely lower accuracy compared to the DML method. Although there is a possibility of improvised accuracy of LBPH by training more data-sets and outstripped camera eminence, it is still impossible to emulate the accuracy of DML using LBPH.

TABLE III
FACE RECOGNITION ACCURACY

Devices	*Recognition Accuracy(%)*	*Webcam*	*Method*
Raspberry Pi	40 - 50 %	Logitech C270	LBPH
CPU	40-50 %	Logitech C270	LBPH
CPU	55 - 65 %	720p HD Laptop	LBPH
CPU	99.38 %	720p HD Laptop	DML

Please note that both of the preliminary results shown in Table II and Table III are based on the simple benchmarks for recognizing just one face in frames of images. If more complicated cases and computations involved, the accuracy and speed will be continually reduced, particularly on the embedded cases. This is one of the motivations of our future work - to offload the complex DML-based computation to the parallel computing devices such as FPGA.

V. CONCLUSION

The preliminary experimental evaluation demonstrates promising results for the proposed architecture with a wide-spread sensor system to cover large places, a scalable security system with face recognition, and an e-mail alert system. In particular, we demonstrate that a wide-range targeted place is able to be covered with BLE Mesh network by implementing a star topology. Additionally, a scalable security system is able to be manipulated by various algorithms, camera quality, hardware's processing speed based on the requirements. For personal uses, the alert system can also be extended to SMS notification depending upon the requirements.

VI. FUTURE WORK

Our future plan is 1) to integrate a Synology cloud server with edge nodes, and with deployed systems in an indirect manner. The Synology server will be used to maintain a webpage for collecting data and making decisions to the network; 2) While the preliminary result shows an evident and robust system, we suspect an immense reduction of latency and a growth in computation speed due to the implementation of cloud and edge computing phenomenon. Thus, in the future, more research on estimating the computation speed on embedded systems and cloud computing will be done, and pure hardware designs with FPGA on high-bandwidth applications and computations such as face recognition will be further applied. 3) We also will provide more benchmarks on evaluating computation time and accuracy such as multiple faces in frames of images. For the low-bandwidth mesh network, further estimation will be provided for a time for executing a task for both cloud and edge computing. Our future goal is to test how many nodes we can add to maximize the distance with acceptable delays to execute a task.

REFERENCES

[1] W. Shi, et al., "Edge Computing: Vision and Challenges," in IEEE Internet of Things Journal, vol. 3, no. 5, pp. 637-646, Oct. 2016.
[2] A. Sajid, et al., "Cloud-Assisted IoT-Based SCADA Systems Security: A Review of the State of the Art and Future Challenges," IEEE Access, vol. 4, pp. 1375-1384, 2016.
[3] L. Nwosu, et al., "Deep Convolutional Neural Network for Facial Expression Recognition Using Facial Parts," 15th IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Feb. 2018.
[4] H. He, et. al., "Dual Long Short-Term Memory Networks for Sub-Character Representation Learning," The 15th Intl. Conference on Information Technology - New Generations (ITNG-2018), Jan. 2018.
[5] A. Gajjar, et al., "An FPGA Synthesis of Face Detection Algorithm using HAAR Classifiers," Intl. Conference on Algorithms, Computing and Systems (ICACS2018), pp.133-137, July 27-29, Beijing China, 2018.
[6] A. Gajjar, et al., "Demo Abstract: A Smart Building System Integrated with An Edge Computing Algorithm and IoT Mesh Networks," The Second ACM/IEEE Symposium on Edge Computing (SEC2017), Article No. 35, Oct. 2017.
[7] X. Yang and X. He, "Establishing a BLE Mesh Network using Fabricated CSRmesh Devices," The 2nd ACM/IEEE Symposium on Edge Computing (SEC2017), Article No. 34, Oct. 2017.
[8] Gareth Halfacree and Eben Upton. 2012. Raspberry Pi User Guide (1st ed.). Wiley Publishing.
[9] Warren Gay. 2014. Raspberry Pi Hardware Reference (1st ed.). Apress, Berkely, CA, USA.
[10] Synology NAS User's Guide,"https://global.download.synology.com/download/Document/UserGuide/DSM/6.2/Syno_UsersGuide_NAServer_enu.pdf"

[11] Bradski, G., “The OpenCV Library,” Dr. Dobb’s Journal of Software Tools, “https://docs.opencv.org/3.4.3/”
[12] Davis E. King., “Dlib-ml: A Machine Learning Toolkit,” Journal of Machine Learning Research 10, pp. 1755-1758, 2009.
[13] Face Recognition Library, “https://pypi.org/project/face_recognition/”
[14] 2015. JM1L2S Hardware Specification. (Aug. 2015). “http://business.aplix.co.jp/ en/product/jm1l2s/”.
[15] 2016. Build a Mesh Network with Bluetooth Low Energy (BLE). (2016). “https: //www.qualcomm.com/products/features/csrmesh”.
[16] Z. Zhang, et. al., “IoT Security: Ongoing Challenges and Research Opportunities,” 2014 IEEE 7th Intl. Conf. on Service-Oriented Computing and Applications, pp. 230-234, 2014,
[17] X. Yang, et. al., “Improving AES Core Performance via An Advanced IBUS Protocol,” ACM Journal on Emerging Technologies in Computing (JETC), Vol. 14, No. 1, PP. 61-63 , Jan. 2018.
[18] X. Yang and W. Wen, “Design of A Pre-Scheduled Data Bus (DBUS) for Advanced Encryption Standard (AES) Encrypted System-on-Chips (SoCs),” The 22nd Asia and South Pacific Design Automation Conference (ASP-DAC2017), PP. 1-6, Chiba, Japan, Feb. 20, 2017.