Design and testing of a dynamic reactive signage network towards fire emergency evacuations

Christopher S. Baidal, Nestor X. Arreaga, Vladimir Sanchez Padilla
ESPOL Polytechnic University, Escuela Superior Politecnica del Litoral, ESPOL,
Faculty of Electrical and Computer Engineering, Ecuador

Article Info

ABSTRACT

The loss of several lives happens in events of natural disasters, due to the strength of nature or the static evacuation routes that usually directs towards a unique exit pathway. This paper proposes the integration of technology tools such as servers and open-source sensors into a reactive signage network to provide an updated and dynamical emergency evacuation system focusing specifically on fire situations. This pilot project works using microcontrollers that capture and manages into a server the surrounding information, setting evacuation routes and behaving according to the Dijkstra algorithm allowing the identification of obstacles or possible incidents that occur spontaneously during an evacuation. This process sends updates to the server either the evacuation route keeps the same or if a modification is required based on the information provided by the sensors about different areas, warning the shortest path for evacuation using computer vision support.

Corresponding Author:

Vladimir Sanchez Padilla,
ESPOL Polytechnic University,
Escuela Superior Politecnica del Litoral (ESPOL),
Faculty of Electrical and Computer Engineering,
Campus Gustavo Galindo Km. 30.5 Via Perimetral, Guayaquil, Ecuador
Email: vladsanc@espol.edu.ec

1. INTRODUCTION

Evacuation systems can be overridden at a moment of panic. Construction regulators ask for buildings with reliable fire alarm systems for a prompt alerting to the people at the time a fire starts, becoming necessary to place pathways with appropriate signage in strategic areas and preset routes. During a catastrophe, people rush to find the first exit available following to other people, without knowing if the pathway ahead is optimal. Besides, the crowding can obstruct the exits causing many people to get trap during chaos [1].

The present proposal integrates different development technologies by four different blocks such as sensors, signage, Dijkstra server, and computer vision, for saving lives according to how fast and guided an evacuation can be handle. We deployed a wireless network to sense an environment, gathering information to establish dynamic escape routes [2] following the premise that authors in [3] mention about that well-established signaling areas are a priority for attaining the evacuation of people if a fire emergency occurs, which lead to the necessity of having updated information during the situation. For our case, the signaling was provided remotely through devices that responded to an evacuation scheme [4] by the integration of microcontrollers as main assets to know the information of the surroundings for forwarding notifications to a gateway. Moreover, the service that manages the signage through the pathways used a designed graph that reacted applying the Dijkstra algorithm to get immediate results and setting values for controlling different routes. Besides, computer vision guaranteed the recognition of obstacles at exits or unexpected incidents that occurred spontaneously during an evacuation [5], supported by data collected from sensors.

We used computer vision for the crowding analysis in the emergency exits and for the physical obstacles that can turn the evacuation delayed and complex [6-9]. Then, people identification was attained by a neuro learning convolutional network [10-12] using two different types of libraries. One of them was TensorFlow, which is useful for predictable frame analysis using all the back images on the learning comparative process. The other one was MatConvNet, useful for head counting on the same image and suitable due to its low response time for a real-time system [11]. Besides, the recognition is a process for the safety of the evacuees, which must not inform the server if the image is a mixed identification schema. Finally, the warnings have to be handle when the process counts several preconfigured poses and heads at the moment of detection.

The rest of this paper is structured as follows: section 2 depicts the deployment scheme with the different parameters that were considered and the system server operation. Section 3 shows the reactive signaling implementation with the messaging information that is needed to differentiate them in the functioning of the network. In section 4, the interpretation of the reactive signaling by the participants is presented with statistics. Finally, section 5 presents the concluding remarks.

2. SCHEMA FOR SENSORS AND SERVERS

A wireless network was deployed to know real-time state changes based on data from sensors. The information monitored from the environment lets to know variable changes that were considered harmful for the people's integrity in each monitored area. Nevertheless, the values to analyze were related to levels of carbon monoxide (CO), carbon dioxide (CO_2), liquefied petroleum gas (LPG), temperature, pressure, and humidity, visualizing their state changes. The values got by the sensing process from the Waspmote PRO V1.5 and Waspmote Gas Sensor Board v 3.0 configured on each sensory node XBee by the ZigBee protocol, allowed the wireless communication between the gateway and the sensors to send the state change to the corresponding gateway for forwarding to the server. The parameters established in the Waspmote gas sensor with the respective values are depicted in Table 1.

Table 1. Waspmote gas sensor structure

Priority	Parameter	Values
High	Carbon Monoxide (CO)	<100 ppm
High	Carbon Dioxide (CO_2)	>350 ppm
High	Liquefied Petroleum Gas (LPG)	>30 ppm
Medium	Temperature	>45 º C
Low	Pressure	>1044 mmHg
Low	Humidity	>50(%)

The sensor network structure included a field gateway to notify the state changes to the server and the continuity of the transfer protocol, especially when implementing a long-range network [13]. Each sensor node was able to do retransmissions bypassing a possible loop in the information conveyed by considering a wireless network deployment, as shown in Figure 1, with XBee PRO S2 antennas, a network coordinator, and routing devices, working with the XBP24-ZB firmware. Both the gateway and the ZigBee protocol coordinator interpreted the information received from the serial interface [14] by software that reset the received messages. Furthermore, a computer node reset the message in communication with the server through a Raspberry Pi 3 Model B, and a handshake system provided through a server establishes the connection. Figure 2 depicts the nodes deployed in communication with a gateway.

The sensor network development must forward information about the state change for the backing up. However, the gateway was in a capacity to understand the message received from the antenna, resetting it from the serial interface by the opening of a COM port through a python program that read the serial communication previously configured in the same way as the antenna. Then, the server sent an alert by a handshake, connecting to a static IP address on an open socket connection for responding to the next step.

For the server system application, two types of plain files identified the number of nodes integrated into the system, such as the edge weights between each other, and the node identifier [15], i.e., the exits between the hall and door department. Moreover, the server started up services that listened to the socket port in which it received the message from the sensor nodes. The identification and the changing state were the principal values to ingress into the service, in addition to the configured identifier with the same department number from the graph. With the support of the modified Dijkstra algorithm [6], the software managed the route among the nodes, providing to all of them the shortest path to the exits. Depending on the exit status, it discarded or updated the routes designed.

Figure 1. Node integrating the gas sensor board and the XBee antenna
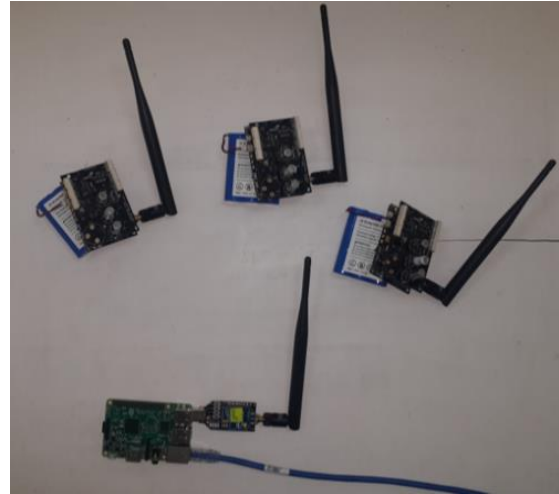

Figure 2. Communication between the nodes and the gateway

After solving the routes, the response generated the identified signals within the different preloaded paths, providing the corresponding values of the state change and the evacuation condition [7], such as signal direction, intensity, and color. Besides, depending on the emergency level, these three parameters indicated the situation for the evacuees, and unexpected alerts from the computer vision front could modify the exit state [16] or the pathway, establishing an update of the exit node into the graph.

3.   REACTIVE SIGNALING SCHEMA

The signaling consisted of a microcontroller that replied to messages received via wireless using the XBee technology. Thus, independently on the sensor state that did the detection and viewing the state information provided by the server about signaling devices as a priority, two types of signaling were considered. The first one, a raster type placed on the floor level. The second one, a panel type in strategic places for visual impact for the evacuees. Nevertheless, the antennas configured by the differentiated network concentrator for the signal message interpretation bypassed failures in the communication between the sensors, turning necessary to transmit the message wirelessly among the Xbee antennas [17-19] through the gateway, including a coordinator and end devices firmware [20-23] for controlling the AT protocol.

The Arduino Uno R3 controlled the signage features [24, 25], such as LED color selector, blinker time, and visual direction through the implementation of an integrated microcontroller, which was able to control the incorporated hardware. The raster message consisted of 8 bits for identifier, 2 bits for coloring, 2 for addressing, and 2 for strobing in the case of raster type; the panel message consisted of 8 bits for identifier, 2 bits for imaging, 2 for addressing and 2 for flashing. Moreover, the server informed the state of the nodes for controlling them through the 2 bits at the control end, i.e., on and off, as shown in Figure 3.

Frame	Id node	Color	Direction	strobe
14 bit total	8	2	2	2

(a)

Frame	Id node	Color	Direction	Figure
14 bit total	8	2	2	2

(b)

Figure 3. Coding frames, (a) for a raster signage message, (b) for a panel signage message

The raster signage was placed at the floor level due to it had to be the first signal that people were able to watch. For the attainment of the luminosity, several segments of LED-strip-RGB control transistors responded to the digital signal alerting to the Arduino based on preloaded arrangements, as shown in

Figure 4. Besides, the stripes outlined the direction depending on whether it was in on or off states, as light strobed reflecting two sets of messages. One, a typebar for loading and unloading. The other was a flashing segment for turning on or off. Test analysis defined responses for a better interpretation, granting the blinker delay among each of the segments and the color configuration as well. This variable was received and interpreted to succeed in the flashing delay time, as shown in Figure 5.
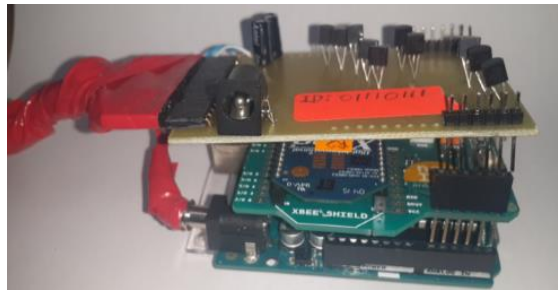

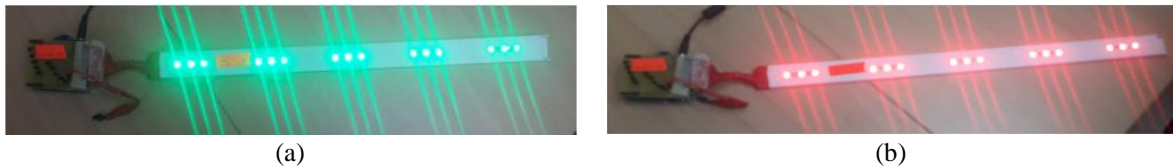Figure 4. Hardware integration of the raster signage

| (a) | (b) |

Figure 5. Raster strip lights representing the direction, (a) green color for enabling an escape route, (b) red color for disabling an escape route

The panel signage incorporated an 8x8 RGB matrix panel wherewith the designed figures were drawn through an arrow either left or right, with an **X** representing a closed path (red) and with an **O** representing an open path (green). This microcontroller configuration discarded message repetitions, not showing the variable, as shown in Figure 6. Moreover, the concentrator had an XBee antenna managed by the ZigBee protocol for the local area network to communicate with the server and receive a message identifier from each node when it changed of state, direction, or even experience intermittency. Nevertheless, it discriminated against the message if the node belonged to a panel type or a raster type, sending the respective message, as shown in Figure 7. At the moment of receiving the result of the server, the coordinator opened a service to listen to the communication socket port for assisting the ID number that needed to change. Once the message received the call, another program opened a serial communication, connecting to an antenna for data transmission to the signage. When the message ID matched the number, the data was processed. Otherwise, it discarded from the buffer of all COM ports.
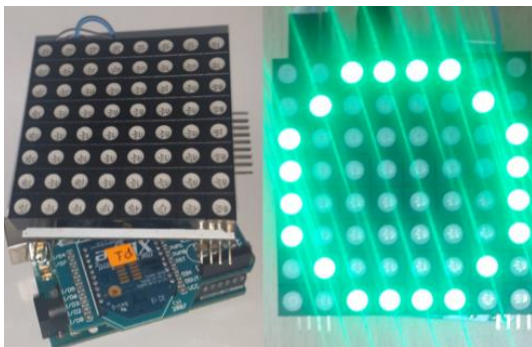

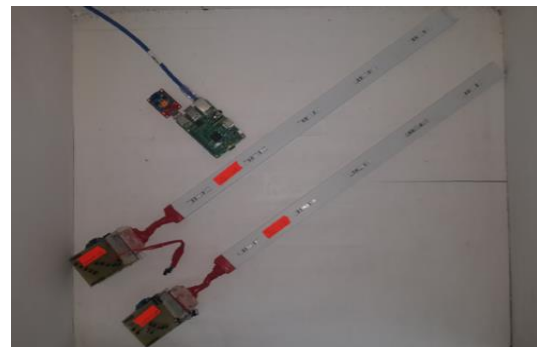Figure 6. Panel signage integration with the XBee shield and RGB 8x8 LED panel

Figure 7. Reactive signage network coordinator and a raster signage

4.   REACTIVE SIGNAL INTEPRETATION

Part of the evaluation process was the analysis of the visual effects in the construction of the reactive signage, organizing a small experiment with outsider people from the investigation development. The tracers and LED screens depicting different orders or instructions were subject to interpretation once they were determined. Initially, five types of sequences were displayed:
-    Lit up of the pads (three led together) with green color shifting to the right, an instruction to follow the right direction.
-    Blinking by the left shift with yellow color for interpreting a caution.
-    Red light flashing for a message of not passing, inducing a search for the next signal.
-    Segmented line cut left, with displacement in the same direction for informing the output is far but enabled to use.
-    Yellow on with an off segment moving to the right, representing caution to the right exit.

One of the objectives was to determine the time in which the participants interpreted the pathway to follow contrasting the exit they followed, whether it was enabled, disabled, or with obstacles. When the sensor light was yellow, people had an error near 35% to identify them. In opposite, when the sensor lights were either green or red, people had no problems identifying them. Figure 8 depicts the sensor color recognition.

Figure 9 shows that sequences 1 and 4 had an error distribution of approximately 15% at the moment of recognizing the signal path, depicting there was no direction interpretation available with no error. Meanwhile, Figure 10 depicts that sequence 4 had better distribution and the estimated average time of approximately 1.85 seconds for recognition. Moreover, the cognitive evaluation determined the real reactions of the participants, such as panic, fear, or stress, with the result of 50% of the recognitions experienced an estimated time between 1 and 2 seconds in sequence 4. The frequency, displacement, and intensity of the RGB LEDs provided timely information regarding the direction and the condition of the evacuation routes. However, the integrated components turned into a guideline for fast evacuation during a fire emergency for solving the routes by established pathways in the software, adding two files for the description.
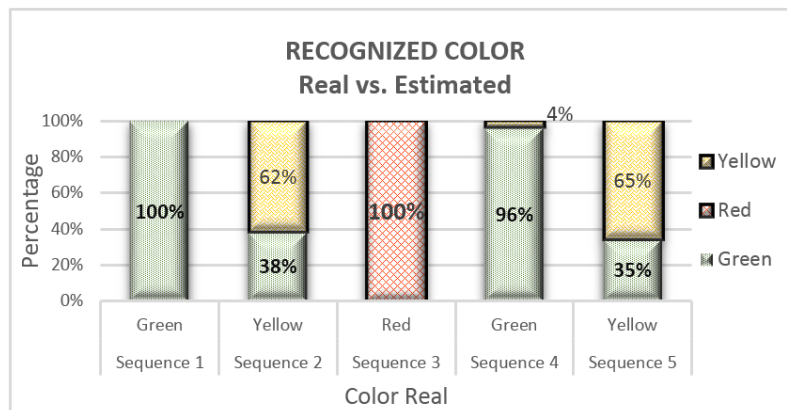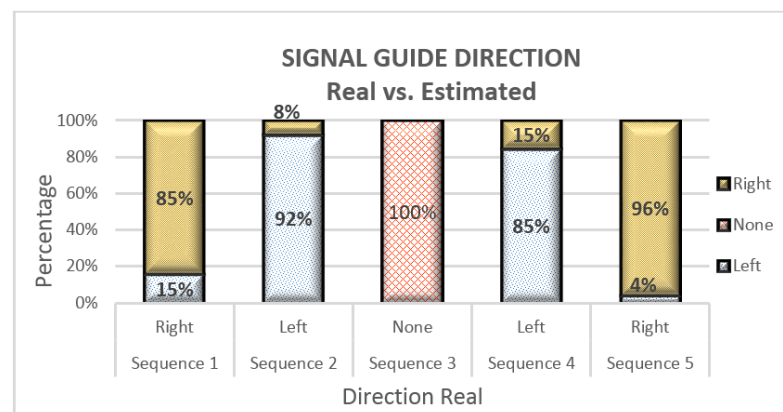

Figure 8. Sensor color recognition

Figure 9. Signal guide direction

Design and testing of a dynamic reactive signage network towards fire emergency ... (Christopher S. Baidal)
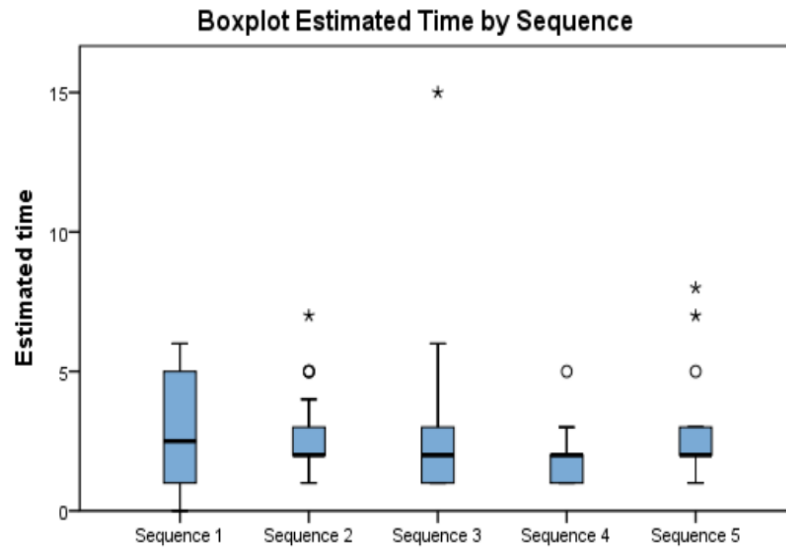
Figure 10. Time sequence distribution

For the testing, the sensors monitored an indoor scenario comparable to a ground floor room of 200 square meters and a height of four meters. The sensors issued an alert when detecting a fire outbreak, sending the node ID to the server as a message 6/1/Description to suggest the best pathway. When the server received the information where a fire outbroke, it started replying to the evacuation route messages for the identification of the raster signage course. Besides, the cameras working as a component of the closed-circuit surveillance asked the connection to the computer vision software to detect unforeseen situations. For its accomplishment, the second camera detected and head counted throughout the same exit, communicating the current state to the server. Finally, the server changed the course of the signage for the evacuees that did not depart, acknowledging different exits in the surroundings that people could handle for a more agile outlet. Figure 11 shows an schematic plot with the best pathway and Table 2 features the ID distribution.
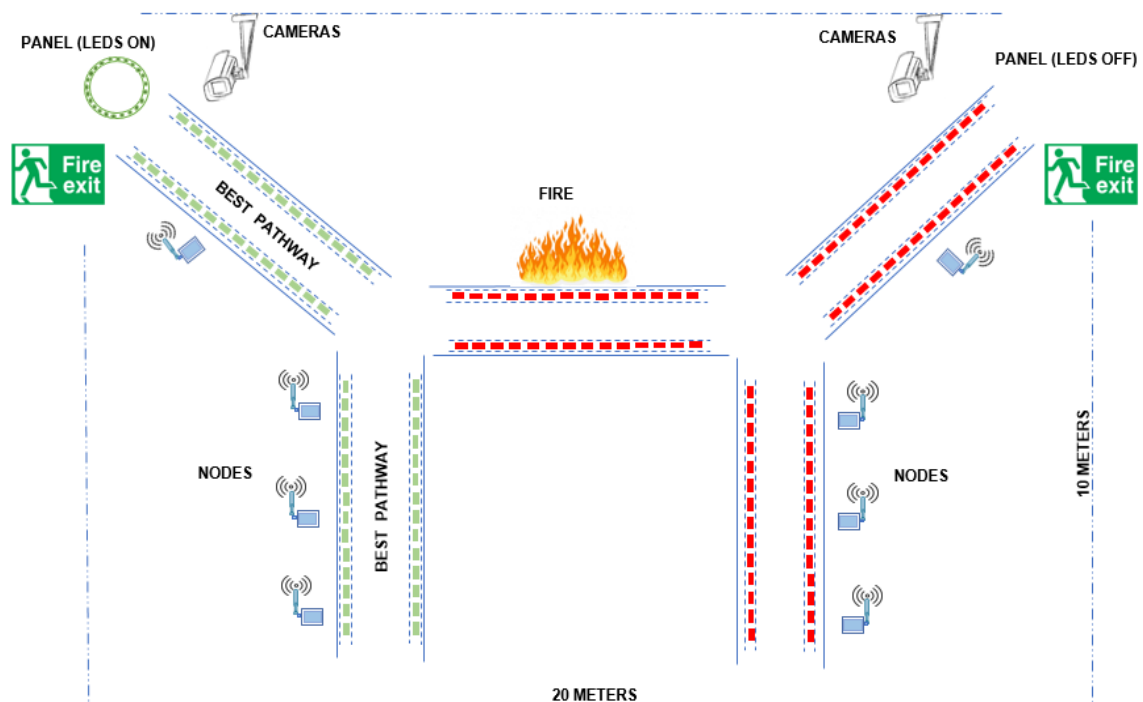

Figure 11. Schematic plot

Table 2. ID desitribution assignation

Node ID	Details
0	Exit 1, Camera 1
1	Exit 2, Camera 2
6	Sensor smoking and temperature node 1
7	Sensor smoking and temperature node 2
8	Signage raster type 1
9	Signage raster type 2
10	Signage panel type 1

5.   CONCLUSION

During the testing, we realized that the deployment of more gateways enhanced the network efficiency and the sensor network response for better performance in case of fire detection. Besides, the Dijkstra server optimized the evacuation route paths for each area, converging the lower level routes into a route presented in the same access. In this study, the signaling guided the participants to faster reaction times to improve the evacuation times. Nevertheless, computer vision plays an important factor in the system due to the recognition features, prioritizing the continuous system training of image detection.

ACKNOWLEDGEMENTS

REFERENCES

[1]   N. Bari, et al., "Internet of Things as a Methodological Concept," *2013 Fourth International Conference on Computing for Geospatial Research and Application*, San Jose, CA, pp. 48-55, 2013.
[2]   M. Barnes, et al., "Emergency Evacuation using Wireless Sensor Networks," *32nd IEEE Conference on Local Computer Networks (LCN 2007)*, Dublin, pp. 851-857, 2007.
[3]   J. A. Munoz, et al., "Adaptive Evacuation Management System Based on Monitoring Techniques," *IEEE Latin America Transactions*, vol. 13, no. 11, pp. 3621-3626, Nov. 2015.
[4]   O. L. Huibregtse, et al., "A Generic Method to Optimize Instructions for the Control of Evacuations," in *IFAC Proceedings Volumes*, vol. 42, no. 15, pp. 217-223, 2009.
[5]   T. Tabirca, et al., "A Dynamic Model for Fire Emergency Evacuation Based on Wireless Sensor Networks," *2009 Eighth International Symposium on Parallel and Distributed Computing*, Lisbon, pp. 29-36, 2009.
[6]   J. Cho, et al., "Application of Dijkstras Algorithm in the Smart Exit Sign," *2014 Proceedings of the 31st ISARC*, Sydney, Australia, pp. 224-229, 2014.
[7]   S. Navarrete and M. Eduardo, "Estudio y simulacion de algoritmos para la evacuacion de personas en situaciones de emergencia sobre una estructura similar al rectorado de la ESPOL," Engineering graduation project, ESPOL Polytechnic University, 2015. Available at: http://www.dspace.espol.edu.ec/xmlui/handle/123456789/30264.
[8]   P. Turaga, et al., "Machine Recognition of Human Activities: A Survey," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1473-1488, Nov. 2008.
[9]   P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, Kauai, HI, USA, pp. 1-1, 2001.
[10] X. Chen, et al., "Pedestrian Detection with Deep Convolutional Neural Network," *Computer Vision, ACCV 2014 Workshops, 12th Asian Conference on Computer Vision, Revised Selected Papers, Part I*, Singapore, pp. 354-365, 2014.
[11] F. Yang, et al., "Exploit All the Layers: Fast and Accurate CNN Object Detector with Scale Dependent Pooling and Cascaded Rejection Classifiers," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, pp. 2129-2137, 2016.
[12] C. Liu, et al., "Raster-to-Vector: Revisiting Floorplan Transformation," *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, pp. 2214-2222, 2017.
[13] S. H. Gajjar, et al., "Wireless Sensor Network: Application led research perspective," *2011 IEEE Recent Advances in Intelligent Computational Systems*, Trivandrum, pp. 025-030, 2011.
[14] F. Ding, et al., "A ZigBee Based Mesh Network for Home Control System," *2008 International Workshop on Education Technology and Training & 2008 International Workshop on Geoscience and Remote Sensing*, Shanghai, pp. 744-748, 2008.
[15] B. Rao and S. Mishra, "An Approach to Detect Patterns (Sub-graphs) with Edge Weight in Graph Using Graph Mining Techniques," *Computational Intelligence in Data Mining, Proceedings of the International Conference on CIDM 2017*, Springer, pp. 807-819, 2019.
[16] B. Kang and H. Choo, "A deep-learning-based emergency alert system," *ICT Express*, vol. 2, no. 2, pp. 67-70, 2016.

[17] I. Calvo, et al., "Building Wireless Control Applications with XBee and LabVIEW," *Applied Sciences*, vol. 9, no. 11, pp. 2379, 2019.

[18] G. L. Ren, et al., "Design and implementation of environmental monitoring using RFID and WSN platform," *2016 IEEE Asia-Pacific Conference on Applied Electromagnetics (APACE)*, Langkawi, pp. 328-333, 2016.

[19] R. Piyare and S. Lee, "Performance Analysis of XBee ZB Module Based Wireless Sensor Networks," *International Journal of Scientific & Engineering Research,* vol. 4, no. 5, 2013.

[20] S. Farahani, "ZigBee Wireless Networks and Transceivers," Elsevier, 2008.

[21] I. Kuzminykh, et al., "Testing of communication range in ZigBee technology," *2017 14th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM)*, pp. 133-136, 2017.

[22] H. Kumbhar, "Wireless sensor network using Xbee on Arduino Platform: An experimental study," *2016 International Conference on Computing Communication Control and automation (ICCUBEA)*, pp. 1-5, 2016.

[23] M. Orgon, et al., "XBee technology: Complex evaluation of radio parameters," *2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, pp. 1-6, 2019.

[24] G. Rajan, et al., "A Review: Comparative Analysis of Arduino Micro Controllers in Robotic Car," *International Journal of Mechanical, Aerospace, Industrial and Mechatronics Engineering*, vol. 9, no. 2, pp. 365-374, 2015.

[25] L. Louis, "Working Principle of Arduino and Using it as a Tool for Study and Research," *International Journal of Control, Automation, Communication and Systems (IJCACS)*, vol. 1, no. 2, pp. 21-29, 2016.