BioGecko

Vol 12 Issue 02 2023
ISSN NO: 2230-5807

Wireless Sensor Networks with Edge Computing for Determination of Dynamic Occupancy of Pre-Determined Space

Kiran Mayee Adavala
Professor in CSE
Vidya Jyothi Institute of Technology
Hyderabad, India
kiranmayee@research.iiit.ac.in

Abstract
Determination of occupancy of a pre-determined space is a requirement for allocation of resources and for several other critical decision-making. The usual method of occupancy determination is a head count. For larger spaces, this requires manual labour and leads to human errors. Nowadays, Sensors are being used in all public space. These sensors can easily be used for determination of occupancy. This paper proposes a design for determination of occupancy with some simple sensors working in sync.

Key Words: Wireless Sensor Networks, ibus, sensors, occupancy, Edge Computing

1. Introduction

Occupancy refers to the number of people placed inside a closed space such as a Bus, Train, Car or any Room such as Conference Hall or Office Space. The number of people who are available or are seated in a closed space can be of paramount importance. For example, if the number of passengers seated in a bus are known, there can be a display outside the bus displaying the number of seats available so that a passenger can avoid boarding a commute that is fully occupied. The same applies to trains as well. Information of this nature can help a commuter take the next available commute. Dynamic Occupancy refers to a scenario where the occupancy value keeps changing over time. It can also be referred to as Temporal Occupancy. The space itself can be stationary – such as a conference hall or mobile such as a train.

Occupancy need not always represent closed spaces. The space can be a Railway platform, a Bus Stop, or any other “not closed” spaces as well. If the number of commuters standing on a platform is known across time for every working day, a pattern can be inferred and a decision can be made on the frequency of trains or buses per hour across the entire day. This information can also be used to make resources such as food, water and air conditioner to be made available at different times throughout the day. For a Conference Hall, occupancy can be used to determine the number of participants in an event.

In an era of autonomous vehicles, determining the number of passengers can be of utmost importance to the intelligent driving system. The position of passengers can also be important in certain cases. Occupancies change in dynamic environment when the mobile system changes state to temporarily reach stationary state.

The sensors that can be used in a bus for determining occupancy is a sensor node cluster working in co-ordination with each other. The algorithm and working of this intelligent sensor node cluster is explained in section 4.

1.1 The ‘ibus’

Hitachi, the Japanese Company has come up with a set of features that can be added to ordinary buses to make them smart. The smart bus, name=d intelligent bus (‘ibus’) [17] has been envisioned to have the following features -

1. Security CCTV Surveillance

This feature involves the use of surveillance technology for safety. It includes Entrance-Exit, traffic and parking surveillance and fire detection.

2. Passenger Experience

This feature is meant for ease of use of bus services. It includes operation, status, congestion mitigation, route search, e-payment and optimization of travel time. This paper deals with determination of number of passengers with the aim of mitigating congestion.

3. Urban Planning

Urban planning has to do with planned urban development, strengthened transport systems, computation of shortest time to destination and planning of cities with increased ease of transportation.

Intelligent Transport Systems

Buses are popular means of transport in cities where there is huge traffic as well as for travel within cities. There are many countries in the world that offer free transportation to public. On the other hand, there are problems like waiting, standing, buses with very less commuters and so on. All these issues point toward the need for a more effective and intelligent solution. In fact, many commuters prefer going by their own vehicles because the commutes are crowded or do not come on time.

The use of personal vehicles leads to roads being fully filled, thereby reducing the speed of vehicles and causing longer commute times for all involved. It also adversely affects the environment. Deteriorating quality of air spaces causes many breathing, skin and heart issues in the long run.

Smart Cities Mission is a Government initiative toward implementing intelligent transport systems that encourage usage of public transport systems. This requires complete data of public transport usage at different points of time during the day, week-end, holidays, day-before-work-day and so on. All these parameters can help create a strong urban transport system.Many companies including Hitachi have taken up this challenge and developed digital solutions that monitor bus operations using Internet of Things (IoT). The data is sent to a central control that updates time of transportation from the commute data. The data is picked from CC TVs, fine and challan collections, GPS and so on.

This helps model new routes, create new commutes, simplify reservations and helps in easing the burden on public. The use of simulators is also becoming popular for demonstrating reduction in traffic congestion and improved business performance, among other key performance indicators (KPIs).

2. Literature Survey

Paper [1] presents a traffic monitoring system implemented through Wireless Sensor Network (WSN) technology within SAFESPOT Project funded by the European Community. The aim of this project is to provide a flexible, robust, low-cost and low-maintenance wireless solution for obtaining traffic-related data that can be used for automatically generating safety warnings at black spots along the road network. The WSN consists of one Gateway Node (GN) and n Sensor Nodes (SNs) deployed along the roadside according to an approximately linear topology. Each SN is able to monitor a section of road, collecting parameters such as vehicle count, speed and direction. Data from the SNs is collected by the GN and delivered to a Road Side Unit (RSU) responsible for fusing it with traffic-related data generated by alternative sources. The system has been tested and performance evaluated under a number of real use-case scenarios.

The rapid progress in the research and development of electronics, sensing, signal processing, and communication networks has significantly advanced the state of applications of intelligent transportation systems (ITSs). However, efficient and low-cost methods for gathering information in large-scale roads are lacking [2]. Consequently, wireless sensor network (WSN) technologies that are low cost, low power, and self-configuring are a key function in ITS. The potential application scenarios and design requirements of WSN for urban transportation (WSN-UT) are proposed in this work. A customized network topology is designed to meet the special requirements, and WSN-UT is specifically tailored for UT applications. WSN-UT enables users to obtain traffic and road information directly from the local WSN within its wireless scope instead of the remote ITS data center. WSN-UT can be configured according to different scenario requirements. A three-level subsystem and a configuration and service subsystem constitute the WSN-UT network frame, and the service/interface and protocol algorithms for every subsystem level are designed for WSN-UT.

Wireless Sensor Network (WSN) plays a vital role in dealing with the challenging tasks of information management in Intelligent Transportation Systems (ITS). Current research has shown that the ever growing number of vehicles on the roads is making congestion worse, and many safety concerns are being addressed comprehensively by the WSN-based ITS [3]. However, the energy consumed by sensor nodes, their operational period, and 'security compromise' are ever growing concerns. Cluster-based routing strategies have potentially contributed in reducing the energy expenditure of sensor nodes, besides the selection of energy-efficient and secure Cluster Head (CH) is still seeking an optimized approach for acquiring the proliferated performance of WSN. To address these concerns, we propose an Intelligent Clustering approach for ITS (ICITS) which selects CHs based on a hybrid optimization method called GABAT that integrates the strengths of Genetic Algorithm (GA) and BAT Algorithm (BA). The proposed framework (ICITS) is targeted primarily to road transport in military areas due to their stringent requirements in terms of security and reliability while collecting the data from the deployed sensor nodes. The simulation results obtained with ICITS demonstrate that it performs well for various performance metrics that include stability period, network survival period and 'number of packets sent', which are improved by 54.7%, 19.6%, and 40.5%, respectively as compared to recently proposed Cluster-based Intelligent Routing Protocol (CIRP).

3. The O-Bus Design

An *Occupany-Knowing Bus* is presented in Figure 1. It has four types of sensors working in co-ordination. All seats have Pressure Sensors. At the opposite sides of the bus at feet level, there are Sonar Sensors. On the inside of the roof, there are Image Sensors, focused on the seats. Finally, there is a CO2 sensor (not seen in the figure) on the inside of the roof that measures CO2 levels inside the closed, Air-conditioned space.

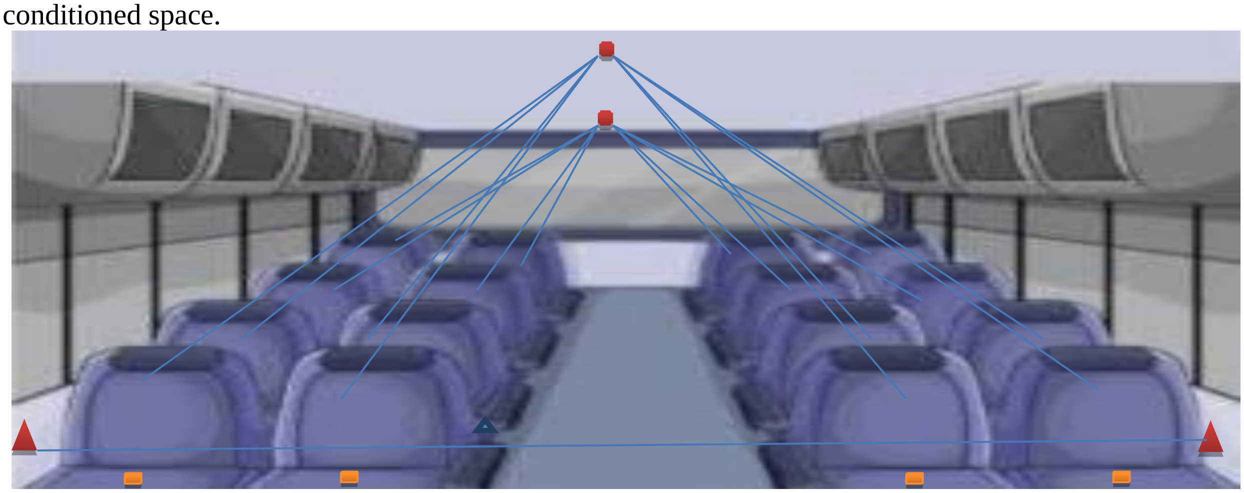

Figure 1. A Space with Occupancy Sensors – red circles – image sensors, red triangles – motion sensors, orange rectangles – pressure sensors. Bus image from https://www.vecteezy.com/vector-art/360721-scene-in-airplane-with-blue-seats

4. Implementation of the Occupancy Determination Module

The assumptions for the rest of this paper are that the closed space dealt with is a bus. The number of seats is 16 as given in Figure 1. There are two seats per row on either side of the bus. The seats on each side are equidistant from each other. The seats are named in matrix order as [1,1], [1,2], [1,3],[1,4] for first row, [2,1],[2,2],[2,3] and [2,4] for second row, and so on. Four sensors are used – Generic (GS), Matrix (MS), Row (RS) and Seat (SS). All outputs except that of Generic Sensors are Boolean with respect to these seat numbers. For example, if Seat Sensor SS4 returns {1}, it means seat [1,4] is occupied. A {0} indicates that the seat is empty. The types of sensors and controller are described in subsections 4.1 through 4.5.

4.1 Generic Sensor (GS)
This is a CO2 sensor that determines the occupancy based on the concentration of CO2 in the pre-determined space. The details of CO2 levels are explained in [ ]. It is assumed that the commute is Air-Conditioned and the windows are therefore not opened by commuters. This assumption is important since otherwise the CO2 level shown would not map to occupancy anymore. There is one Generic Sensor, which we name GS1 in our case study.

4.2 Matrix Sensor (MS)
This is an Image sensor that determines if the seats in its purview are occupied or not. In this discussion, we assume that each Image sensor senses occupancy in 8 seats just below it as show in Figure 1. Also, the range of the sensors are mutually exclusive – i.e., if sensor A picks data with respect to rows 1 and 2, no other Image sensor will pick the same data. There are two Matrix Sensors MS1 and MS2 in our case study – MS1 picks image signals from seats in row 1 i.e., {[1,1],[1,2],[1,3],[1,4]} and row 2 i.e., {[2,1],[2,2],[2,3],[2,4]}. Similarly, MS2 picks images from seats in row 3 and row 4. In this work we represent MS1 output as {0,0,0,0,0,0,0,1} to represent an occupancy at [2,4]. MS2 output is also represented in a similar fashion.

4.3 Row Sensor (RS)
This is a motion sensor and determines if the row in its purview is occupied. Each row has two motion sensors fixed to both bus walls at foot level. The two motion sensors are programmed to pick signals only from the two seats next to it. Therefore, all motion sensors are also mutually exclusive. There are four rows in our case study. Therefore, there are 8 motion sensors placed pairwise at opposite corners from row 1 to 4 and named RS1 through RS8. The first row sensors i.e., RS1 senses inputs from seats [1,1] and [1,2]. RS2 which is opposite to RS1 senses motion with respect to seats [1,3] and [1,4]. The same logic applies to the remaining three rows in the bus. The output of row sensors is represented as , for example {0,1}

4.4 Seat Sensor(SS)
This is a pressure sensor placed inside seats and determines if the seat is occupied. Needless to say, every seat has this sensor placed below the cushion. The sensor not only determines occupancy, it also specifies approximately if the commuter is a child or a grown up, given the range of weights for each of these categories. There are 16 sensors, one for each seat labeled SS1 through SS16. Each of these sensors gives two inputs – whether seat [x,y] is occupied or not and an approximation of whether the occupant is a child or an adult.

4.5 Edge Computing - Controller
This is a simple micro-controller device programmed to receive inputs from the four sensors and to locally produce a single, integrated output that maps to all inputs. It also makes decisions in the face of conflicting inputs as well. The controller initializes all seats to {0} to indicate all seats are empty. Every time there is a change in occupancy, The controller updates the seat status.

4.6 Digital Display Device
The controller is connected to a simple digital display board that is placed outside the bus. This displays the current occupancy of the bus.
A 2D layout design of all sensors is given in Figure 2. It shows a single Generic Sensor (GS), two Matrix Sensors (MS), eight Row Sensors (RS) and sixteen Seat Sensors (SS). The sensors in the middle – GS,

MS1 and MS2 are placed on the inside of the roof . An example for the outputs of the various sensors and controller deduction is given in Figure 3.

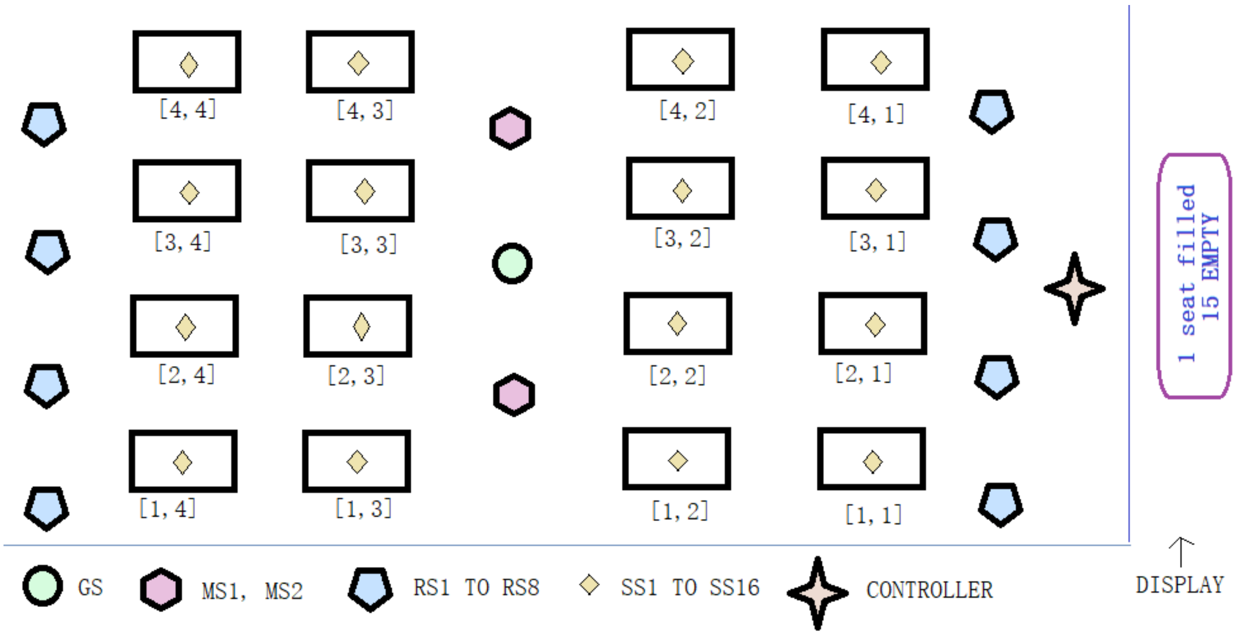

Figure. A 2D layout of the four sensors in a closed, pre-determined space.

5. Discussion

An output from the sensor cluster and the controller deduction is given in Table 1. There is a conflict in sensor outputs, with Row Sensor showing two occupants in fourth row at seats [4,1] and [4,2]. The controller compares this observation with the triangulated outputs of all other sensors and concludes that there is an error in this output and there is exactly one occupant at seat [4,1].

Table 1. A sample output expected from the sensor design.

Device Id (me)	**Sensor ID**	**Output**	**Controller Deduction**
GS001	GS	290	1 Occupant
MS001	MS1	{0,0,0,0,0,0,0,0}	No occupant in first two rows
MS002	MS2	{0,0,0,0,1,0,0,0}	One occupant in third and fourth row - probably in [4,1]
RS001	RS1	{0,0}	No occupant in first row on LHS
RS002	RS2	{0,0}	No occupant in first row on RHS
RS003	RS3	{0,0}	No occupant in second row on LHS
RS004	RS4	{0,0}	No occupant in second row on RHS
RS005	RS5	{0,0}	No occupant in third row on LHS
RS006	RS6	{0,0}	No occupant in third row on RHS
RS007	RS7	{1,1}	Two occupants in fourth row on LHS
RS008	RS8	{0,0}	No occupant in fourth row on RHS
SS001	SS1	{0}	No occupant in seat [1,1]
SS002	SS2	{0}	No occupant in seat [1,2]

SS003	SS3	{0}	No occupant in seat [1,3]
SS004	SS4	{0}	No occupant in seat [1,4]
SS005	SS5	{0}	No occupant in seat [2,1]
SS006	SS6	{0}	No occupant in seat [2,2]
SS007	SS7	{0}	No occupant in seat [2,3]
SS008	SS8	{0}	No occupant in seat [2,4]
SS009	SS9	{0}	No occupant in seat [3,1]
SS0010	SS10	{0}	No occupant in seat [3,2]
SS0011	SS11	{0}	No occupant in seat [3,3]
SS0012	SS12	{0}	No occupant in seat [3,4]
SS0013	SS13	{1}	Occupant in seat [4,1]
SS0014	SS14	{0}	No occupant in seat [4,2]
SS0015	SS15	{0}	No occupant in seat [4,3]
SS0016	SS16	{0}	No occupant in seat [4,4]

Conflict Resolution

	RS	Two Occupants in fourth row
	SS	One occupant in fourth row in seat [4,1]
	MS	One occupant in third and fourth row - probably in [4,1]
	GS	1 Occupant
	Precedence RS<GS<MS<SS	
	Conclusion	*One occupant in fourth row in seat [4,1]*

The general Algorithms for Controller Logic and Sensor Logic are given in Figures 3 and 4respectively. The controller logic runs until bus is in parked( ) state. It calls every sensor for changed inputs. Having received inputs from all sensors, it runs a conflict-resolution algorithm for triangulating the results obtained from all the sensors. The results are sent to “display” device.

The algorithm for sensor logic shows that the sensor senses change in occupancy for the seat(s) in its purview every 30 seconds via the Sense_Next( ) function. Whenever the status changes the changed_status( ) is True. The sensor sends back this data to Controller. If there is no change, the sensor loops until there is a change in occupancy status.

6. Limitations

Sensors may miss the number of people who stand since they are all focused on the seats. This can be because the number of buses or trains is less at that time. CO_2 sensor readings may approximately give the total occupancy in such cases. There is also a possibility of people placing baggage in their adjacent seats giving the devices an impression that seats are filled.

Conclusion

This paper deals with the application of IoT towards determining the dynamically changing occupancy of a space. The concept is explained with the case study of a Bus. The various sensors that are required are explained. The sample outputs from the various sensors and a resolution of conflicting sensor inputs is presented. Finally, generic algorithms for the controller and sensor are presented. Future work in this area is the simulation of the presented design.

```
Algorithm Controller_Logic(GS, MS, RS, SS)
Begin
     Start_all_Sensors()
     //run forever in steps of 30 seconds
     for time=0 in steps of 30 seconds do
     Begin
          if parked()
          break;
          //GS is a number - CO2 level
          GS = Generic_Sensor_Logic()
          //MS1 amd MS2 are 2D matrices
          MS1 = Sensor_logic("MS001")
          MS2 = Sensor_logic("MS002")
          for i = 1 to 8 do
          Begin
              str = "RS00"+i
              RS[i] = Sensor_logic(str)
          End
          for i = 1 to 16 do
          Begin
             str = "SS00"+i
             SS[i] = sensor_logic(str)
          End
          Resolve_Conflict()
          display(occupancy)
     End
     Stop_all_Sensors()
End.
```

Figure 3. Algorithm for Controller Logic

```
Algorithm Sensor_Logic(ID)
Begin
while (True)        // loop forever
     Begin
          Sense_Next(ID)
          if changed_status(ID)
               return (Sensor_Data)
     End
End.
```

Figure 4. Algorithm for Sensor Logic (common to all sensors)

References

[1] M. Franceschinis, L. Gioanola, M. Messere, R. Tomasi, M. A. Spirito and P. Civera, "Wireless Sensor Networks for Intelligent Transportation Systems," VTC Spring 2009 - IEEE 69th Vehicular Technology Conference, 2009, pp. 1-5, doi: 10.1109/VETECS.2009.5073915.
[2] X. Hu, L. Yang and W. Xiong, "A Novel Wireless Sensor Network Frame for Urban Transportation," in IEEE Internet of Things Journal, vol. 2, no. 6, pp. 586-595, Dec. 2015, doi: 10.1109/JIOT.2015.2475639.
[3] A.P. Singh, V. Jain, S. Chaudhari, F. A. Kraemer, S. Werner and V. Garg, "Machine Learning-Based Occupancy Estimation Using Multivariate Sensor Nodes," 2018 IEEE Globecom Workshops (GC Wkshps), Abu Dhabi, United Arab Emirates, 2018, pp. 1-6, doi: 10.1109/GLOCOMW.2018.8644432.
[4] Yuan Gao, Yew-Soon Ong, Maoguo Gong, Fenlong Jiang, Yuanqiao Zhang, Shanfeng Wang, "Breaking Hardware Boundaries of IoT Devices via Inverse Feature Completion", IEEE Internet of Things Journal, vol.10, no.6, pp.5519-5529, 2023.

[5] E Ramanujam, Arpit Sharma, J Javid Hussian, Thinagaran Perumal, "Improving Indoor occupancy estimation using a hybrid CNN-LSTM approach", 2022 International Conference on Intelligent Controller and Computing for Smart Power (ICICCSP), pp.1-6, 2022.
[6] Deepa R, K.Mohan Raj, N. Balaji, Durgadevi K, "Machine Learning based Estimation of Room Occupancy Using Non-Intrusive Sensors", 2022 International Conference on Communication, Computing and Internet of Things (IC3IoT), pp.1-5, 2022.
[7] Jun Jiang, Chenli Wang, Thomas Roth, Cuong Nguyen, Patrick Kamongi, Hohyun Lee, Yuhong Liu, "Residential House Occupancy Detection: Trust-Based Scheme Using Economic and Privacy-Aware Sensors", IEEE Internet of Things Journal, vol.9, no.3, pp.1938-1950, 2022.
[8] Alexia Dini Kounoudes, Georgia M. Kapitsaki, IoannisKatakis, MariosMilis, "User-centred privacy inference detection for smart home devices", 2021 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/IOP/SCI), pp.210-218, 2021.
[9] Mina Younan, Mohamed Elhoseny, Abd El-Mageid A. Ali, Essam H. Houssein, "Data Reduction Model for Balancing Indexing and Securing Resources in the Internet-of-Things Applications", IEEE Internet of Things Journal, vol.8, no.7, pp.5953-5972, 2021.
[10] Muhammad Emad-Ud-Din, Ya Wang, "Indoor Occupancy Sensing via Networked Nodes (2012–2022): A Review", Future Internet, vol.15, no.3, pp.116, 202
[11] Pragati Kumari, Priyanka Kushwaha, Muskan Sharma, Pushpanjalikumari, Richa Yadav, "Home Occupancy Estimation Using Machine Learning", Advanced Network Technologies and Intelligent Computing, vol.1798, pp.522, 2023.
[12] Alberto Amato, Vincenzo Di Lecce, "Data Analysis for Information Discovery", Applied Sciences, vol.13, no.6, pp.3481, 2023.
[13] Jehyun Kim, JongIl Bang, Anseop Choi, Hyeun Jun Moon, Minki Sung, "Estimation of Occupancy Using IoT Sensors and a Carbon Dioxide-Based Machine Learning Model with Ventilation System and Differential Pressure Data", Sensors, vol.23, no.2, pp.585, 2023.
[14] Krati Rastogi, Divya Lohani, "Edge Computing-Based Internet of Things Framework for Indoor Occupancy Estimation", Research Anthology on Edge Computing Protocols, Applications, and Integration, pp.619, 2022.
[15] Matin, M. A., & Islam, M. M. (2012). Overview of Wireless Sensor Network. InTech. doi:10.5772/49376.
[16] Dr. Kiran Mayee Adavala. (2023). Determination of Best Non-Intrusive Sensors for Dynamic Occupancy Detection. *Scandinavian Journal of Information Systems*, *35*(1), 976–985. Retrieved from http://sjisscandinavian-iris.com/index.php/sjis/article/view/420.
[17] https://social-innovation.hitachi/en-in/knowledge-hub/viewpoint/intelligent-transportation-systems/
[18] www.researchgate.net