iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 Issue: VI Month of publication: June 2022

DOI: https://doi.org/10.22214/ijraset.2022.45122

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

Room Occupancy Prediction Using ML

Dr M Trupthi
IT Department, Chaitanya Bharathi Institute of Technology

Abstract: Now-a-days buildings are growing rapidly and uses large amount of energy. Most of the energy gets wasted on HVAC systems. Occupancy (presence and number of occupants) is one of the most important factors impacting energy efficiency of HVAC systems. Hence, knowing occupancy information is vital for demand driven HVAC controls, that directly impacts on energy-related building control systems. So, occupancy information without privacy issues (like using cameras) gained importance.

Recent works are done on this problem. Existing solutions proposed are yes or no classification, estimating for low count etc using sensors data. Using yes or no classification, the energy consumption may not be saved to a great extent. In this work, we have trained a model which classifies the state of room based on number of occupants (empty, low, fair, high). For this we are using data collected by different sensors (co2, humidity, temperature). We are using classification algorithms to train the model. This information could be used to solve energy related problems in buildings.

Keywords: Multiple Heterogeneous Sensor Nodes, Decision trees, KNN, SVM, Random Forest, Occupancy prediction

I. INTRODUCTION

Real-time occupancy data will cause intelligent heating, ventilation, and air conditioning (HVAC) and lighting systems in buildings, which could not only save energy but also improve occupant comfort. With the introduction of the online of Things (IoT), there are now readily accessible sensors which can quantify environmental parameters, and this data are often analyzed using machine learning (ML) to assess human occupancy without the use of video-based systems. Recent studies have shown that buildings with established occupancy patterns can save 30% electricity.

In the early stages of occupancy detection and estimation, invasive systems like cameras, WiFi, wearables, and RFID were used. With growing privacy concerns, researchers have turned their attention to the utilization of nonintrusive environmental sensors for occupancy detection and estimation, like CO2, temperature, light, motion, and humidity. We used subsequent three readily available low-cost sensors for occupancy estimation because the strain of this project is additionally on the utilization of non-intrusive sensors: CO2, temperature, humidity.

So occupancy estimation is that the quantitative sensing of humans present in spaces. Real- time analysis of occupancy can decrease energy consumption through better allocation of building resources.

Real-time occupancy information could also be a serious input for automating heating, ventilation and air conditioning (HVAC) and lighting systems in buildings which could not only conserve energy, but also provide better comfort to the occupants. With the arrival of Internet of Things (IoT), there are readily available sensors which can measure the environmental parameters and this data are often analyzed using machine learning (ML) to figure out human occupancy without video based systems. Past research shows that having smart building management with occupancy, can reduce building energy consumption by 30 – 50%. Regarding the occupancy estimation, several works have been carried out based on one or more environmental variables. However, few works that did not use non environmental variables as additional support for the estimation were found.

On the one hand, there are works, such as Adeogun et al. used pressure, humidity, and CO2, in addition to a Passive Infrared (PIR) sensor. A second example is the work of Chitu et al. , which in addition to using CO2, also registers the state of all the airflow sources, obtaining an accuracy of 0.69. On the other hand, among the works that only use environmental variables, Jiang et al. and Zhou et al. were found. Both used CO2 to estimate occupancy, obtaining an accuracy of 0.77 and 0.82, respectively. Another example is the work of Viani et al , which obtained an accuracy of 0.82 using temperature, humidity, and CO2. It is relevant to mention that there are no research works based only on environmental variables that do not use CO2. Furthermore, none of the studies based only on environmental variables presents an accuracy similar to that obtained by works using environmental and non-environmental variables, such as Adeogun et al .

In this paper aims to estimate the occupancy level during a room by using multiple heterogeneous sensor nodes with various ML techniques like K-Nearest Neighbours(KNN), SVM (Linear), random forest (RF), Decision trees(DT).

II. LITERATURE SURVEY

A lot of research has been administered within the literature for occupancy detection, i.e., if the space is occupied or not. Although detection can help in improving energy savings, estimating the precise number of occupants can make the system even more energy-efficient. a typical implementation is to use RFID tags, as seen during a paper by Li et al in 2012 during this paper, the researchers have readers and reference tags placed around a neighborhood a tracking tag placed on any human within the world. A notable paper by Abade et al. places multiple sensor nodes that detect noise to count the quantity of humans present. While these designs work for one room, they're too difficult to implement in large settings like office space buildings. Another pattern of labor is based upon computer vision concepts. Companies like VergeSense install a camera per room to detect occupancy. While such an implementation limits hardware to a minimum of one device per room, it still requires meticulous setup and raises privacy concerns.Three ML techniques namely Hidden Markov model (HMM), artificial neural network (ANN) and support vector machine (SVM) were used on a distributed sensor network. it had been shown that HMM gives the simplest performance with 75% accuracy. In this paper, we aim to estimate the number of occupants in a room by using heterogeneous sensor nodes with various ML techniques like SVM, random forest(RF), decision trees(DT).

Table I Existing System

Solution	**Met**thodology	Drawbacks
IoT-based occupancy monitoring techniques for energy-efficient smart buildings (2015)	Most building entrances and infrastructure objects have video cameras, these devices and others help foster occupancy detection.	• The quality depends on light conditions. • Easily used for user identification or privacy violation
Occupancy detection from electricity consumption data (2013)	A home's pattern of electricity usage generally changes when occupants are present due to their interaction with electrical loads.	• Insensitive when people do not use electricity or some other goods.
A Machine Learning Approach to Indoor Occupancy Detection Using Non-Intrusive Environmental Sensor Data(2019)	Uses temperature and humidity data collected from sensors and applying machine learning algorithms .	• Accuracy is low and will get effected when room space is increased.
Ensemble-based extreme learning machine model for occupancy detection with ambient attributes(2020)	Used temperature, humidity, hum. ratio, CO2, light to detect occupancy.	• detection alone can help in improving energy savings, estimating the precise number of occupants can make the system even more energy-efficient.

III.PROPOSED METHODOLOGY

To build our system we first decided upon the sensors we might use. A 2013 study by Yang et al. Investigated multiple off-the-shelf sensors and mapped the correlation between their measurements and human presence. They demonstrated that, a wellplaced light sensor, CO2, temperature, and humidity showed most variance compared to multi-human occupancy. We decided to use the latter three because the light sensor's performance was dependent upon specific placement, which might have limited deployment ease. We are Estimating for comparatively large occupants and dividing it to 4 groups:

Group - Range Empty (0) – 0
Low (1) - 1, 2
Fair (2) - 3, 4
High (3) - 5, 6

A. Algorithm

1) *Step 1:* A delay of 200 seconds is maintained to read the sensor data values.
2) *Step 2:* When the condition holds true:The data values of various sensors are collected and kept being updated for every 200 seconds.
3) *Step 3:* And then a connection is established to cloud (api.thingspeak.com) when the connection is not interrupted, the sensor values are pushed to cloud with adelay of 15 seconds
4) *Step 4:* The values are then shown in the form of seperate fields (named as field1, field2...).
5) *Step 5:* When the condition holds true: Failed to read from sensors is displayed with a warning note that somewhere the connection has lost.

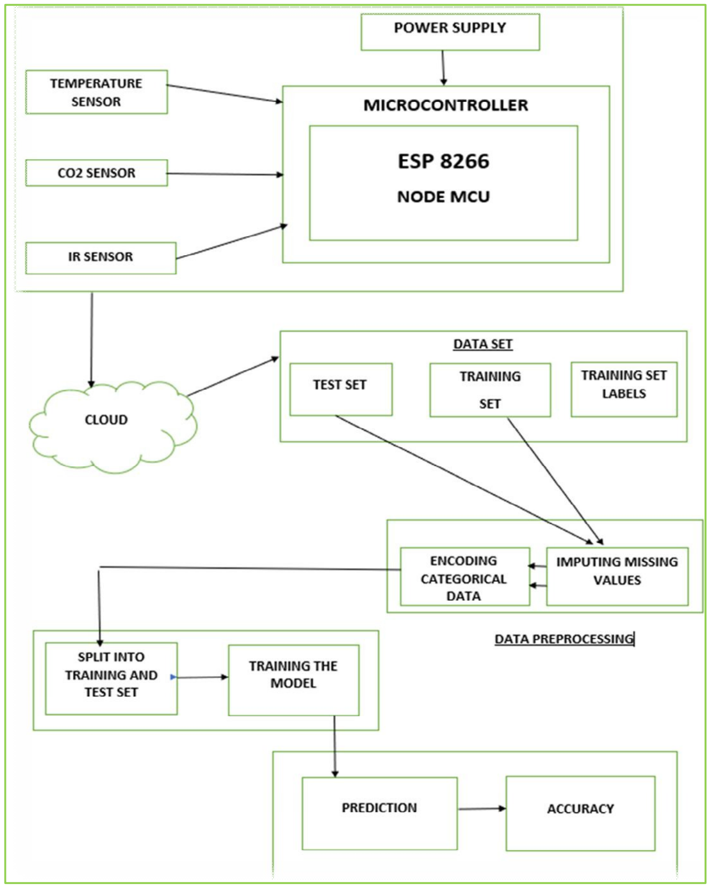


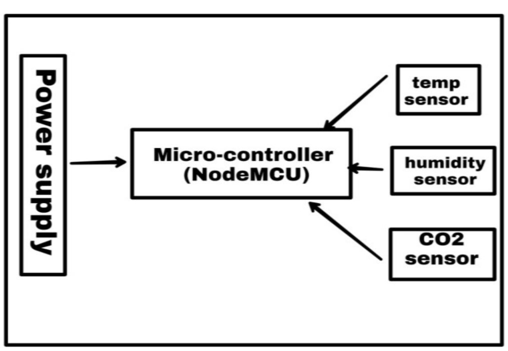
Fig. 1 System Design

IV. IMPLEMENTATION

A. Phase 1: Setting up Sensor Node

To build our system we first decided upon the sensors we would use. A 2013 study by Yang et al. investigated multiple sensors and mapped the correlation between their measurements and human presence. They demonstrated that, in order, a well-placed light sensor, CO_2, temperature, and humidity showed most variance compared to multi-human occupancy. We decided to use the latter three because the light sensor's effectiveness was dependent upon specific placement, which would have limited deployment ease.

Our system prototype consisted of a NodeMCU. We integrated a *SEN0159* CO_2 sensor using a along with an *DHT22 Temp+Humidity* sensor. Given that our raw data would be 4- dimensional (time, CO_2, temperature, humidity). we've deployed our prototype during a medium sized room. the info for our machine learning structure was collected during a room over 3000 minutes. Above mentioned sensors give accurate measurement upto 6 decimals.



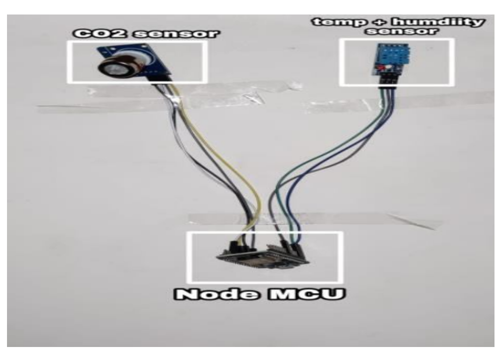


Fig. 2 Sensor Node Setup

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

B. Phase 2: Pushing data into cloud

ThingSpeak is an Open-Source IoT application and API to store and retrieve data from Hardware devices and Sensors. It uses HTTP Protocol over the Internet or LAN for its communication. The MATLAB analytics is included to analyze and visualize the data received from your Hardware or Sensor Devices.

The Read API key allows you to read data from a private channel. You can find the Read API key for a channel on the API Keys tab of your ThingSpeak channel view. If you are reading data from a public channel, you do not need a Read API key. Save your channel Read API key in a variable for convenience.

Pushed the data collected using the sensor node made in the above phase into cloud. And retrieved data into a csv file.

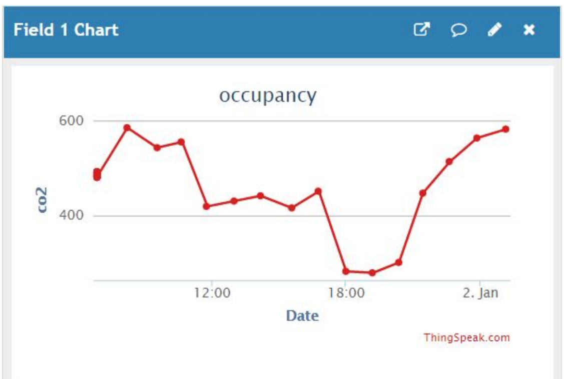


Fig. 3 CO_2 vs Time Data

Fig. 4 Temperature vs Time Data

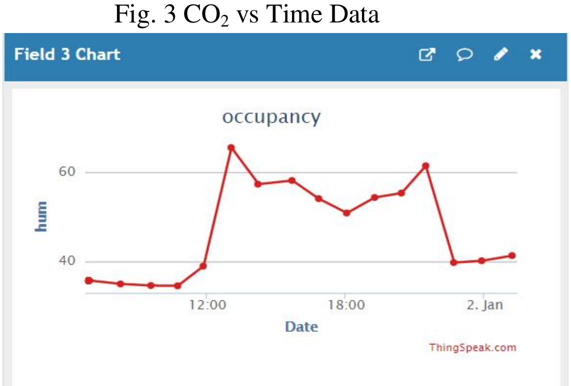


Fig. 5 Humidity vs Time Data

Fig. 6 Number of Occupants vs Time

C. Phase 3: Data Preprocessing

Dataset Construction

The data from all sensor nodes was not sent at the same time. There was a few seconds of variance in the arrival times of sensor data. We created a popular stamp by combining the time stamps within a specified time interval. We have over 7000 instances of data collected every 30 seconds. Rows with missing data were removed from the table. As you can see we noted down the reading with different number of occupants in the room. So according to the readings which we have noted we had written occupants count in the occupants column of the dataset. In this way we had created our dataset. After constructing the dataset we split the dataset manually into test and training data. Test dataset is used for making the predictions and training data is used for training the MachineLearning model.

All sensor nodes did not send their data at the same time. A few seconds of variation was present between the arrival times of data of sensors. We merged the time-stamps within a given time interval into a common stamp. We have collected data for each 30 seconds andhave over 7000 instances. Rows with missing data were deleted. For occupants count, the inhabitants in the home kept a personal track of every time they entered and left the livingroom. After the training data was collected, these logs and saved images were manually studied to count the number of humans present in the living room for each sensor data entry. The ground truth of occupancy count was appended. We have appended a new occupancy level column based on number of occupants dividing it into empty(0), low(1), fair(2), high(3).

	temp	hum	occ	CO2	occupancy_level
0	32.6943	38.84717	5	485.5357	3
1	32.69651	38.8195	5	485.659	3
2	32.69861	38.79287	5	485.7701	3
3	32.70063	38.76728	5	485.8695	3
4	32.70255	38.7427	5	485.9576	3
5	32.70437	38.71911	5	486.0348	3
6	32.70611	38.6965	5	486.1014	3
7	32.70776	38.67484	5	486.158	3
8	32.70933	38.65413	5	486.205	3
9	32.71081	38.63433	5	486.2426	3
10	32.71221	38.61544	5	486.2715	3
11	32.71354	38.59744	5	486.292	3
12	32.71479	38.5803	5	486.3044	3
13	32.71596	38.564	5	486.3094	3
14	32.71707	38.54854	5	486.3071	3
15	32.7181	38.53389	5	486.2982	3
16	32.71907	38.52003	5	486.283	3
17	32.71997	38.50695	5	486.2619	3
18	32.72082	38.49462	5	486.2354	3
19	32.7216	38.48303	5	486.2038	3
20	32.72232	38.47217	5	486.1677	3

Fig. 7 Sample Data of Dataset

D. Phase-4(Applying Machine Learning Algorithms)

Classifying the occupancy state into 4 groups, multi-class classificationalgorithms are best suited to train the data.

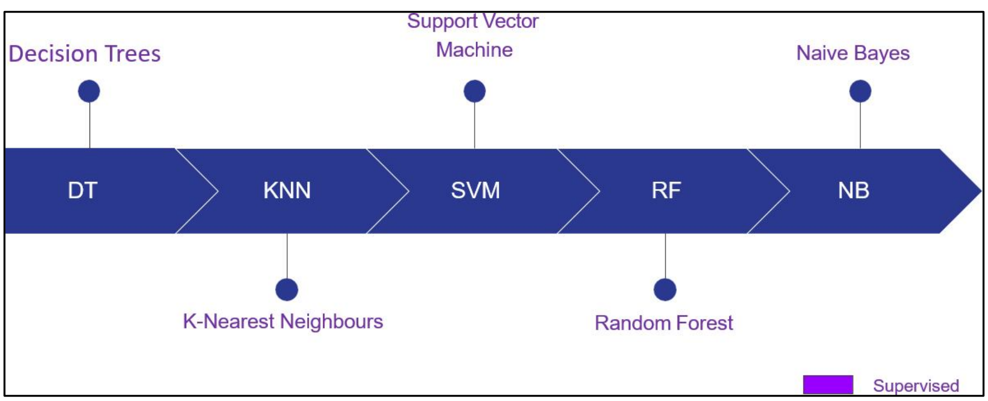


Fig. 8 Algorithms

V. RESULTS AND ANALYSIS

The ML algorithms discussed within the previous section were implemented using Scikit-learn [6]. Metrics like accuracy, F1 score and confusion matrix were calculated. Since the info is of time-series nature, data wasn't Randomised before cross validation to avoid data points almost like test data stepping into the training data. needless to say, the entire dataset with all the sensors, performs the best at estimating the level of occupants accurately. KNN performed well among all. We got accuracy of 87.12% and f1-score (86.8%). Except Naïve Bayes, remaining algorithms got an accuracy above 80%.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

Table II Performance of algorithms

Algorithm	Accuracy	F1-score
KNN	87.12	86.86
SVM	80.7	80.2
Decision Trees	84.55	84.4
Random Forest	85.1	84.9
Naive Bayes	60	59

A. *Confusion Matrix of Best Performed Algorithm*

K – Nearest Neighbours performed well against all other algorithms and gave an accuracy over 87%. Below is the confusion matrix of KNN.

It describes the performance of a classification model. It shows relation between true values vs predicted values of test dataset. Overally 226 wrong predictions were made out of 1788. If we take this for Occupancy detection (yes or no), only 48 wrong assumptions were madewhich gives us about 99.88% accuracy for occupancy detection.

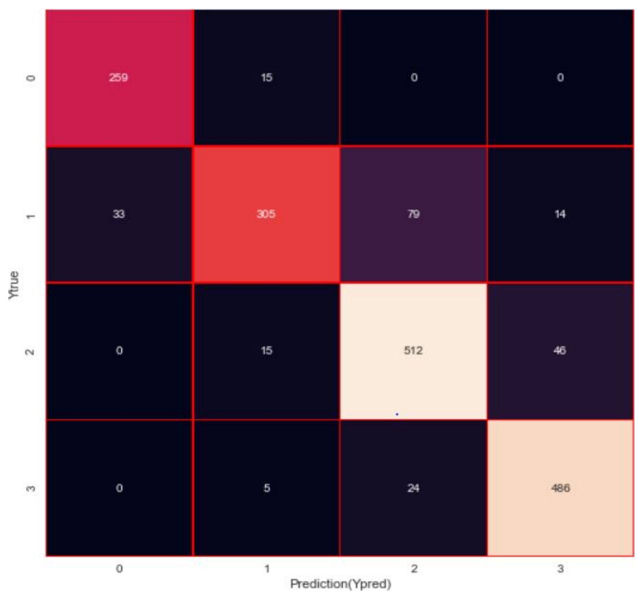


Fig. 9 Confusion matrix of KNN

VI. CONCLUSIONS AND FUTURE SCOPE

While occupancy analytics may be a growing application domain, current sensing solutions are lacking thanks to their inability to scale to realistic environments and encroachment on privacy. This motivated the event of a non-intrusive occupancy estimator that attempts to attenuate scaling costs and maximize data privacy. A prototype was built using an embedded controller which used CO2, temperature and humidity. we got an accuracy of 85%. This paper provided a functional verification of the feasibility of a non-intrusive occupancy estimator, however didn't improve the system and software architecture. This leaves numerous avenues to be taken that might further the event of such a tool . Notably, in terms of the system architecture, more sensors are often carefully examined and added so as to extend the dimensionality of the input data.

The experiments in this work were conducted in a small room. We plan to extend this model to large workspaces in the future. Certain derived features like time and type of day can also be taken into account provided our dataset is large and spans multiple weeks. We also plan to conduct real-time experiments in the near future. In order for the a system to determine the occupancy in a room, it would have to sense the environment, analyze the data (i.e. detecting and counting people), generate an answer, and make the answer available in a suitable format. In other words, it has to detect and count the number of occupants in a room and communicate this to a context broker. In this thesis the occupancy sensor system should send notifications of changes in occupancy indicating that there are zero, one, or many person/persons in a room.

REFERENCES

[1] L.M. Candanedo, V. Feldheim, Accurate occupancy detection of an office room from light, temperature, humidity and CO2 measurements using statistical learning models, Energy Build. 112 (2016) 28–39. doi:10.1016/j.enbuild.2015.11.071.

[2] T. Hong, H.-W. Lin, Occupant Behavior: Impact on Energy Use of Private Offices, in: Asim 2012, 1st Asia Conf. Int. Build. Perform. Simul. Assoc., 2013.

[3] N. Li, G. Calis, B. Becerik-Gerber, Measuring and monitoring occupancy with an RFID based system for demand-driven HVAC operations, Autom. Constr. 24 (2012) 89–99. doi:10.1016/j.autcon.2012.02.013.

[4] H. Rafsanjani, C. Ahn, M. Alahmad, A Review of Approaches for Sensing, Understanding, and Improving Occupancy-Related Energy-Use Behaviors in Commercial Buildings, Energies. 8 (2015) 10996–11029. doi:10.3390/en81010996.

[5] Chen, Z.; Jiang, C.; Xie, L. Building Occupancy Estimation and Detection: A Review. Energy Build. 2018, 169, 260–270. [CrossRef]

[6] Erickson, M. A. Carreira-Perpinan, and A. Cerpa. 2011. OBSERVE: Occupancy-based system for efficient reduction of HVAC energy. In Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN'11). ACM/IEEE, 258–269.

[7] Advanced occupancy sensing for energy efficiency in office buildings,Tobore Ekwevugbe, Neil Brown, Vijayanarasimha Pakka, Denis Fan, published in Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering.

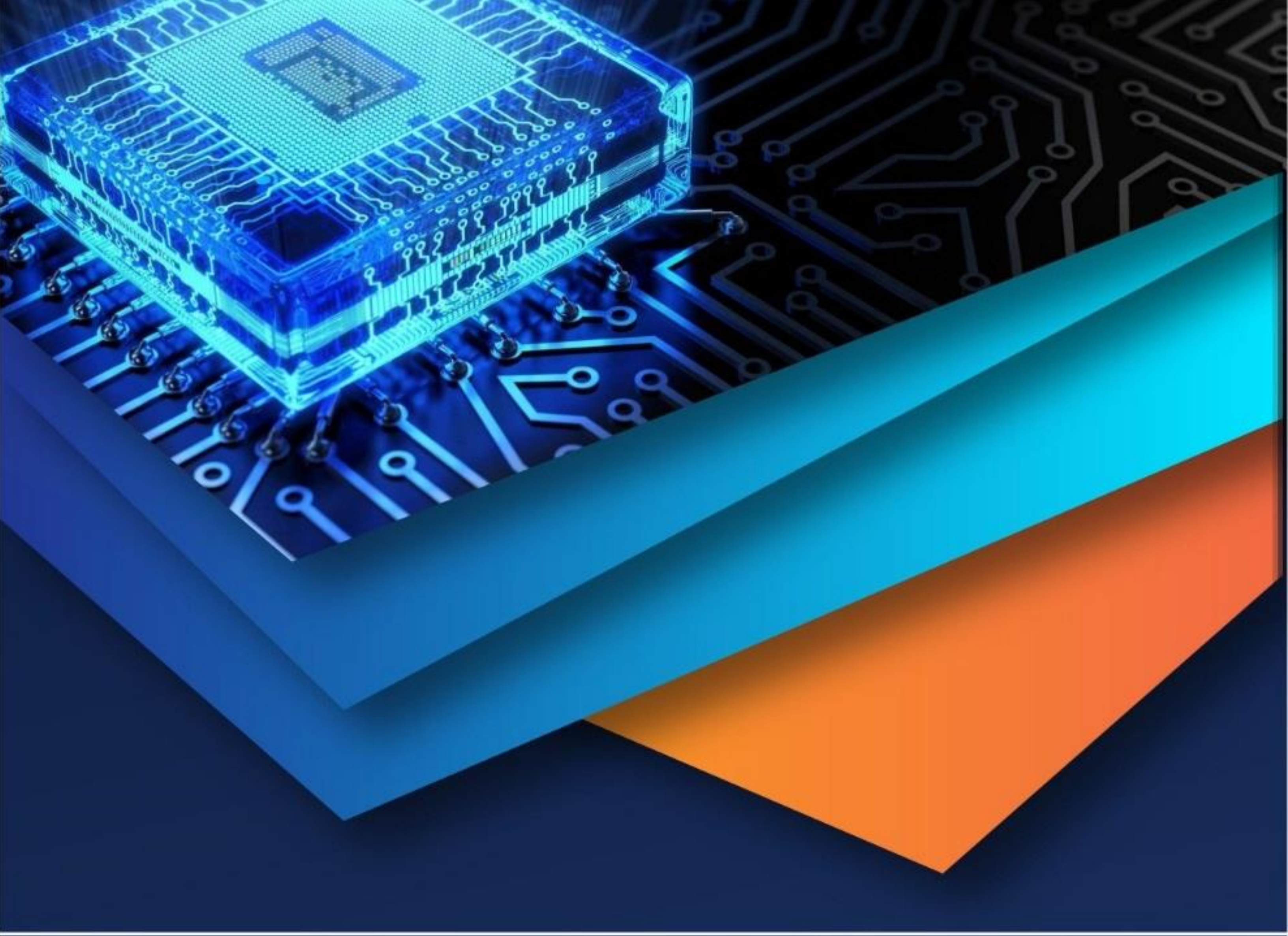
[8] Occupancy detection in commercial buildings using opportunistic context sources, Sunil Kumar Ghai ; Lakshmi V Thanayankizil ; Deva P. Seetharam ; Dipanjan Chakraborty, published in Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on 19-23 March 2012

[9] Z. Yang, N. Li, B. Becerik-Gerber, M. Orosz, A systematic approach to occupancy modeling in ambient sensor-rich buildings, Simulation vol. 90, Issue 8, pp. 960–977, August 2014.

[10] Vigna, I.; Balest, J.; Pasut, W.; Pernetti, R. Office Occupants' Perspective Dealing with Energy Flexibility: A Large-Scale Survey in the Province of Bolzano. Energies 2020, 13, 4312.

[11] Q. Zhu, Z. Chen, M.K. Masood, Y.C. Soh, Occupancy estimation with environmental sensing via non-iterative LRF feature learning in time and frequency domains, Energy Build. 141 (2017) 125–133. doi:10.1016/J.ENBUILD.2017.01.057.

[12] T. Hong, Y. Chen, Z. Belafi, S. D'Oca, Occupant behavior models: A critical review of implementation and representation approaches in building performance simulation programs, Build. Simul. 11 (2018) 1–14. doi:10.1007/s12273-017- 0396-6.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY