Application Oriented Sensor Database System

G S R Satyanarayana, Debarka Chakraborty, Santos Kumar Das*
Department of Electronics and Communication
National Institute of Technology Rourkela, INDIA
* Email:dassk@nitrkl.ac.in

***Abstract*—Sensor network, which is actually a collection of spatially separated sensor nodes, forms an integral part of smart city applications. The sensors used in the network must be interconnected with each other and must possess communication and data processing capabilities. Thus every layer of the city is being impacted. The data received from the sensors must be stored in the database and analyzed for different applications. The applications include traffic congestion management, fire detection, healthcare application and so on. This led to the evolution of sensor database system. In this paper, the interfacing of sensor using different communication technology such as serial communication, Ethernet communication etc. is being discussed. A client-server model is developed in case of Ethernet communication through socket programming done in python. The data obtained from the sensor is stored in the MySQL database through the python script running in the personal computer (PC). Finally, the data obtained from the sensors is analyzed for a fire detection application process. The sensors used for this purpose are DHT11 sensor and MQ2 gas sensor.**

***Keywords*—serial communication, Ethernet communication, socket programming, UDP, DHT11 sensor, MQ2 sensor, SIM900 GSM module.**

I. Introduction

Nowadays, physical world is being transformed into computing platform due to the availability of small scale sensors, actuators and embedded processors. Sensor network is basically a combination of large number of sensor nodes. These sensor nodes possess physical sensing capabilities such as temperature, pressure, light etc. together with networking and computation capabilities. These sensor nodes must also possess communication capability. It finds application in large number of areas such as military, surveillance, healthcare environments etc. [1]. To measure noise, temperature, light etc. of any office or building, sensor network can be extremely useful. There are many factors which influence the design of sensor network design such as scalability, fault tolerance, cost, power constraint, hardware constraint etc. [1], [2]. For routing in the sensor network, energy efficiency and scalability are the key features [3]. SPIN and directed diffusion are amongst the data-centric routing protocols whereas LEACH, PEGASIS belongs to hierarchical protocols and GAF AND GEAR belongs to location based routing protocols used in sensor networks [3], [4], [5]. Now, secure transmission of information over wireless sensor network has also to be ensured. Major attacks to wireless sensor network correspond to Denial of Service (DoS), Sybil Attack, Hello Flood Attack, Wormhole Attack, Blackhole Attack [6], [7]. Different schemes are also available to provide security against these attacks [6].

Now, for extracting sensor data most of the existing applications depend on a centralized method. The centralized method for extracting data from sensors has many drawbacks which is overcome by the evolution of sensor database system [8], [9]. TinyDB is a query processing system which is used for extracting information from wireless sensor network [10], [11]. This system supports declarative query processing. Spatial data types and spatial operators are not supported by TinyDB. So, for both spatial and aspatial query processing in sensor nodes SE TinyDB was designed [12], [13], [14]. Execution time and memory usage are better for spatial TinyDB as compared to TinyDB but TinyDB outperforms spatial TinyDB in terms of accuracy [12]. The architecture of a sensor database system consists of a separate query plan at the source sensor and a query plan at the leader node [15]. In-network Aggregation is utilized in the distributed system where part of the computation is pushed inside the sensor network and this significantly extend the lifetime of the network [15], [16], [17], [18].

This paper deals with the interfacing of the sensor to the MySQL database using different communication medium and analyzing the sensor data stored in the MySQL database for a fire detection application. Early days fire detection has done by visual inspection but after the discovery of the fire and smoke sensors scenario has been changed. Recent works show how a fire detection can be effectively done with a variety of methods, some of them are sensor based [19] and some are done with the help of image processing [20], [21], [22]. the cost of the image processing equipment is more they can be easily detecting the fire and smoke. They are most suitable for outdoor applications, since they can observe more area, but for indoors they may not suitable due to their cost. The alert mechanisms after detection of fire accidents are done with audio, SMS etc. This model can be used for other applications without changing the major architecture of the system. one of the major challenges in fire detection is to reduce false alarms, we check validity of the sensor readings to reduce false alarms [22].

The paper has been organized in the following way. Section II comprises the description of smart city network model. Section III explains the interfacing of sensor to the MySQL database using different communication medium. In section IV Based on proposed network model an application for fire detection has been developed, where the process of the data stored in the MySQL database has been shown. Lastly, the

conclusion is presented in section V.

II. Smart City Network Model

A smart city is a place which utilizes information and communication technology to provide better services and better communication. Here, the essential services are provided in a faster, flexible and efficient manner with the aid of Internet of Things (IoT). Smart cities are no longer a concept of future. In fact, with the fast expansion of Internet of Things (IoT) and the way it is impacting municipal services all over the world, smart cities are growing rapidly. A smart city makes use of sensors, transducers, desired technology for the connection of every component across the city. In fact, sensors form the basic building block of a smart city network. The data collected from the sensors are processed, stored in the database and analyzed for different applications. Thus, every layer of a city is being impacted starting from the street lights to the air that human beings are breathing. Real time responses are allowed in a smart city application. Thus, it is capable of handling more challenges as compared to that of a normal city. Thus, urban life becomes more efficient, environment friendly and resources are being utilized in a proper manner. The quality of urban life is improved to a greater extent benefitting everybody starting from the citizen to businessman. The technologies of a smart city must include smart energy, smart transportation, smart data, smart infrastructure, smart mobility, smart IoT devices etc. Various nations around the world have been captivated by the smart city concept. In fact, many countries have started to invest heavily for the smart city research purpose. Nowadays, India is also planning for smart cities. Thus, sensor data management has emerged as a promising field of research. The schematic of a smart city network is shown in Fig. 1.

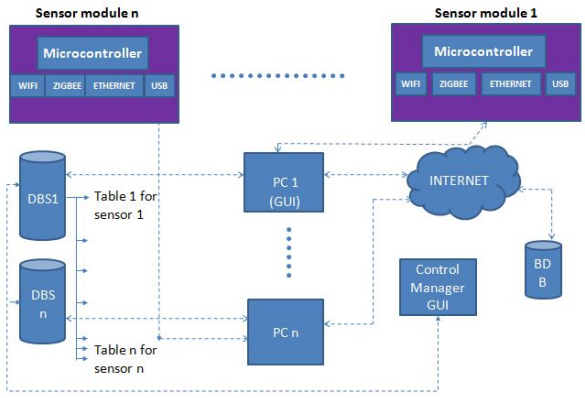


Fig. 1. Schematic of a Smart City Network

a) : The basic features of the network shown in the above figure are as follows:

i. Here, a microcontroller is being interfaced with every sensor.
ii. Accessing of the microcontroller can be done through various communication mediums such as serial communication, Ethernet, Wi-Fi, ZigBee etc.
iii. The data collected from the sensor can be stored in its respective database which can be analyzed for different applications.
iv. The status of each small database is being looked upon by a control manager GUI.
v. Lastly, all the respective small databases are combined into a central big database.

III. Interfacing of Sensor using different Communication Medium

A. Serial Communication

The basic block diagram for interfacing a singular sensor to the MySQL database is shown in the Fig. 2.

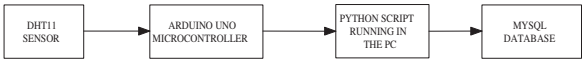


Fig. 2. Interfacing of DHT11 sensor to MySQL database

Following are the points in the serial interfacing of sensor shown above:

i. The sensor used is the DHT11 sensor which gives the temperature and humidity of a particular area.
ii. Arduino UNO is used as the microcontroller board or kit for this interfacing purpose.
iii. The python scripts are running in the PC, which can be used for collecting the data from the DHT11 sensor through the arduino microcontroller and storing the same in the MySQL database.
iv. Lastly, the database handling operation is being done by MySQL.

In this interfacing, the data is being received by the Arduino UNO as the digital input from the DHT11 sensor. The data received is being stored in the respective variables. The data is now being checked for validity according to the valid range which the DHT11 sensor measure as given above. In the program, to check for the validity an integer variable is taken and if measured sensor values are within the allowable range as mentioned above, then the integer variable outputs its value as 1, otherwise it gives the value as 0. The serial monitor output is shown in the following Fig. 3 The python programs which are running on the PC are used to perform all the data manipulation operations on the already setup MySQL database. The python programs are used for table creation in the MySQL database, inserting data which is being read from the sensor into the table already present in the MySQL database. The weather data in the MySQL db is shown in the Fig. 4. A python program can also display the sensor data which is being stored in the MySQL database in the python shell itself. The display of weather data in the python shell is shown in Fig. 5.

B. Ethernet Communication

Here, a client-server model has been developed. The client PC sends request over the network to the server arduino. The server receives the request, performs the desired operation and

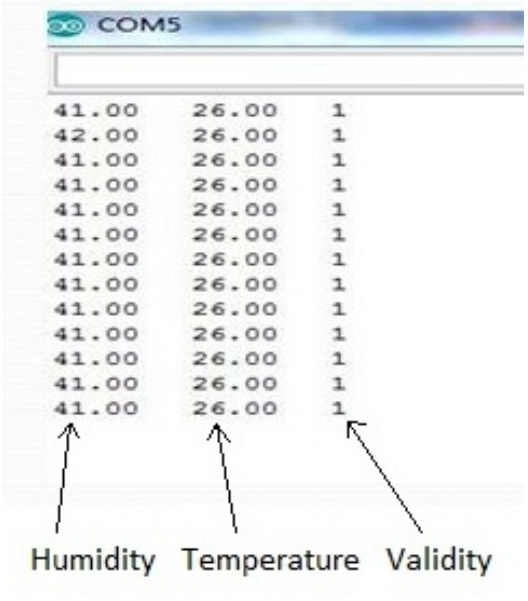


Fig. 3. Output values in the serial monitor

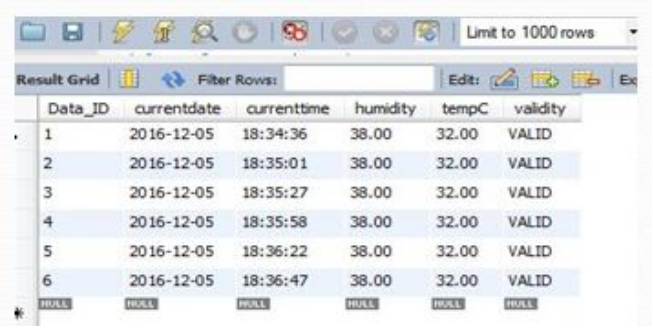


Fig. 4. Display of table weather in the MySQL workbench

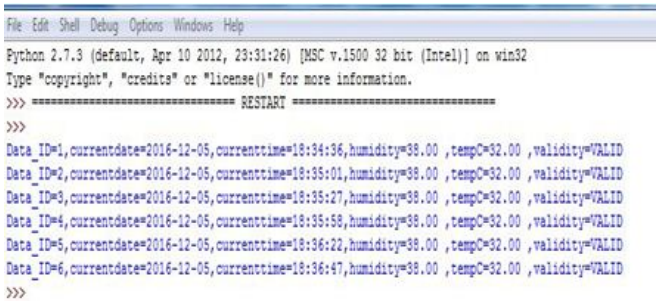


Fig. 5. Display of weather data in the python shell

sends the result of the operation back to the client which is waiting for a reply. In this case, the server arduino sends the temperature and humidity of an area, collected from the DHT11 sensor connected to it, back to the client PC after receiving a request from it. The Ethernet interfacing of a sensor with the MySQL database is shown in Fig. 6. Network communications that are being done by PC are performed by

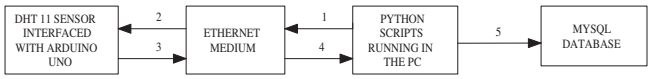


Fig. 6. Ethernet interfacing of a sensor with MySQL database

means of socket. Socket supports bidirectional communication in network programming after the successful establishment of sockets in both the server and client side. The communication between sockets can be done within the same process, or within processes on the identical devices, or within processes on the different devices. Socket Programming using UDP is being implemented here. The serial monitor output is shown in the Fig. 7. The python script running on the client PC is

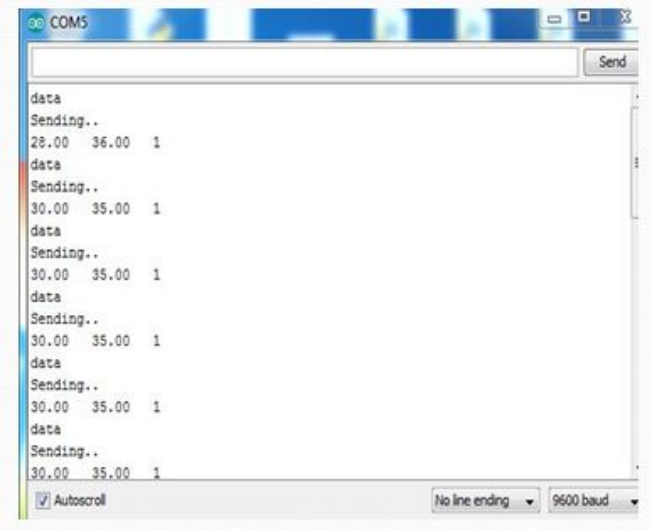


Fig. 7. Serial Monitor Output

responsible to perform all the data manipulation operation starting from collecting the sensor data from an Ethernet interfaced microcontroller to updating the sensor data to an already set up MySQL database. In the first implementation case, the python script was receiving the sensor data over the USB serial port and updating it to the database. But, now it is receiving the sensor data over Ethernet. The flowchart of the python script for collecting the sensor data from the Ethernet interface and updating to the MySQL database and the output display of the table in the MySQL workbench are shown in Fig. 8 and Fig. 9.

C. Problems encountered

Certain problems have been encountered in the process of data transfer over Ethernet by means of UDP transport layer protocol. These are explained as follows:

i. **Unusual behaviour of UDP sockets**
The UDP socket is seen to perform unusually in windows 10 platform although it was performing quite accurately in windows 7 platform. The main reason behind the problem is the unavailability of the data from the arduino

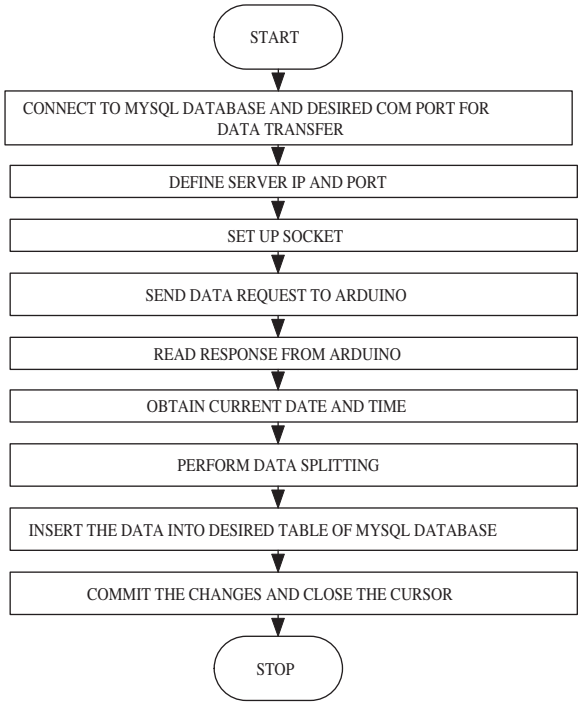


Fig. 8. Flowchart for updating the data obtained over Ethernet to MySQL database

Result Grid | Filter Rows: | Edit:

Data_ID	currentdate	currenttime	temp	humidity	validity
1	2017-01-27	18:46:43	29.00	34.00	VALID
2	2017-01-27	18:48:09	29.00	34.00	VALID
3	2017-01-27	18:48:50	30.00	34.00	VALID
4	2017-03-11	19:56:19	28.00	36.00	VALID
5	2017-03-11	20:01:05	30.00	35.00	VALID
6	2017-03-11	20:02:42	30.00	35.00	VALID
7	2017-03-11	20:03:02	30.00	35.00	VALID
8	2017-03-11	20:04:33	30.00	35.00	VALID
NULL	NULL	NULL	NULL	NULL	NULL

Fig. 9. Display of table in the MySQL workbench

in the socket buffer, although the python program running on the client PC is able to send request to the server arduino through the same socket.

ii. **Abnormal execution of sql queries**
To update the data in the MySQL database, insert operation is being used. Now, this operation is found to perform quite accurately when smaller number of fields is to be inserted into the database but it fails when the number of fields increases gradually. No logical reason is found for such anomalous behaviour .

IV. Fire Detection Application

Proposed application has been developed using DHT11 sensor, MQ2 gas sensor, arduino UNO board and a GSM module. The temperature and the humidity of a particular place are given by the DHT11 sensor whereas MQ2 gas sensor gives the status of gas leakage in a particular area. LPG, hydrogen, smoke etc. are detected by MQ2 gas sensor. A threshold value is set relating to the temperature measured by DHT11 sensor and output voltage of the MQ2 gas sensor. If the measured value is greater than the threshold value then there is a confirmation of fire detection and a SMS alert will be sent and the sensor data will be updated to the MySQL database. Thus, along with the development of the fire detection application its data is also linked with the MySQL database. The required hardware connection for the developed fire detection application is shown in the following Fig.10 The

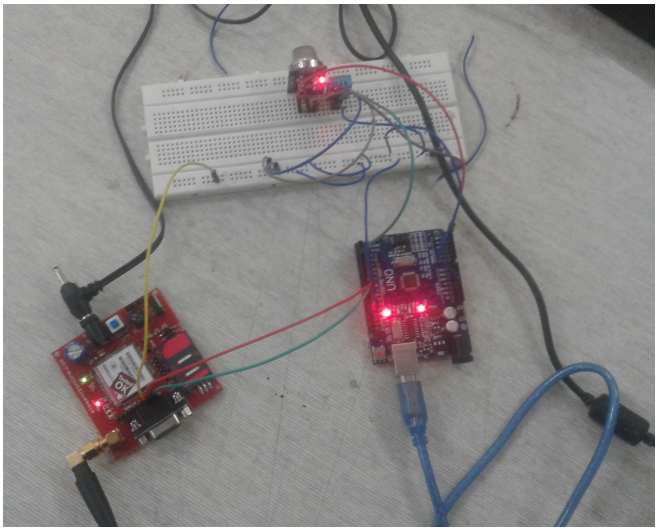


Fig. 10. Hardware connection for the fire detection application process

flowchart for such an implementation is given below Fig.11. The Fig. 12 shows the SMS alert that has been sent at a particular instant of time on the detection of fire. The data that is updated to the already created table in the MySQL database by the python script running on the PC is shown in the following Fig. 13. The table is having six fields as DataID, current date, current time, tempC as the temperature measured by the DHT11 sensor, voltage which is the output voltage as measured by the MQ2 gas sensor and the status field which gives the status of fire detection.

V. Conclusions

As smart city is no longer a concept of future, sensor data management has been emerged as a promising field of research nowadays. Sensor network forms the backbone of a smart city. Sensor nodes in a wireless sensor network must communicate with each other. The data from the sensor network must be stored in a database which can be analyzed for different applications. The technologies of the smart city include smart

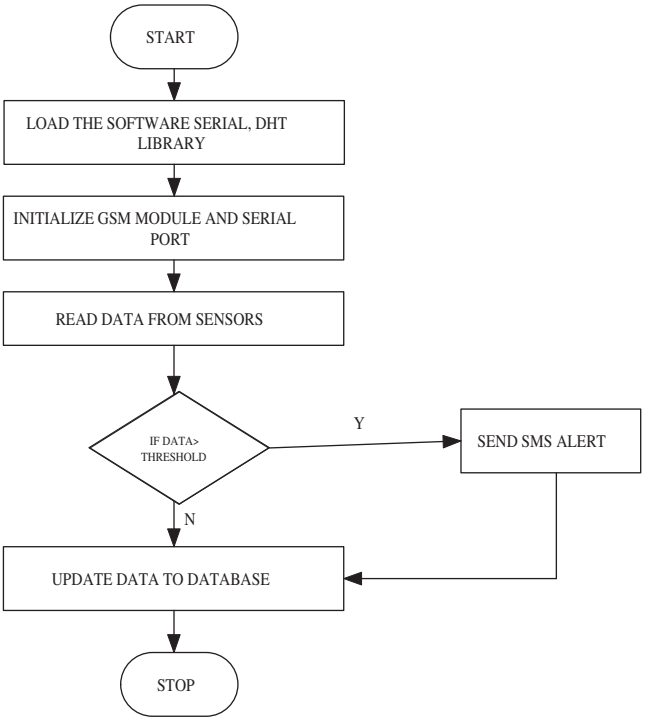


Fig. 11. Flowchart for the fire detection application process

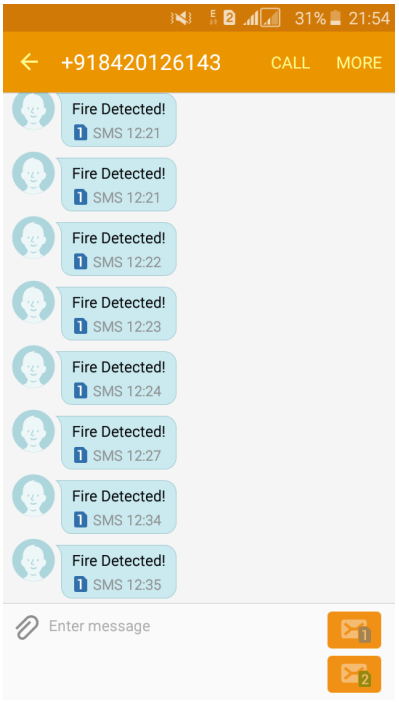
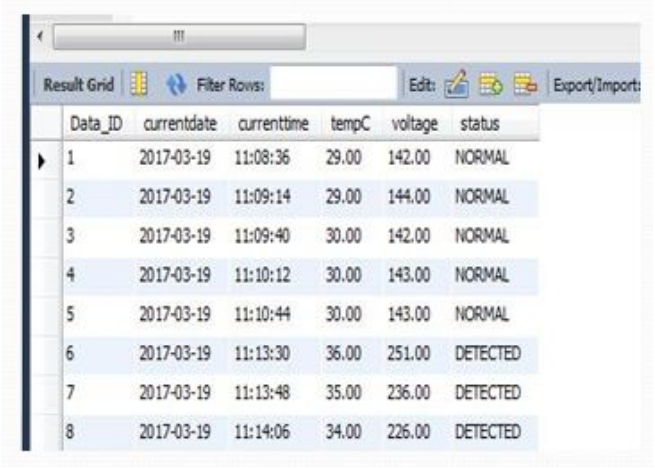


Fig. 12. SMS alert on the detection of fire

energy, smart transportation, smart data, smart IoT devices etc. sensor database system forms an integral part of smart city. This paper actually deals with the storing of the sensor



Data_ID	currentdate	currenttime	tempC	voltage	status
1	2017-03-19	11:08:36	29.00	142.00	NORMAL
2	2017-03-19	11:09:14	29.00	144.00	NORMAL
3	2017-03-19	11:09:40	30.00	142.00	NORMAL
4	2017-03-19	11:10:12	30.00	143.00	NORMAL
5	2017-03-19	11:10:44	30.00	143.00	NORMAL
6	2017-03-19	11:13:30	36.00	251.00	DETECTED
7	2017-03-19	11:13:48	35.00	236.00	DETECTED
8	2017-03-19	11:14:06	34.00	226.00	DETECTED

Fig. 13. Display of data in the MySQL database

data in the MySQL database and analyzing the stored data for different applications. With this purpose in view, at first the interfacing of sensor to the PC through microcontroller is being done using different communication medium. Now, the sensor data stored in the MySQL database is analyzed for a fire detection application process which can be used to detect the occurrence of fire in the surroundings. The fire detection application has been developed by using DHT11 sensor, MQ2 gas sensor, Arduino UNO and SIM900 gsm module.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, no. 4, pp. 393–422, 2002.

[2] V. Potdar, A. Sharif, and E. Chang, "Wireless sensor networks: A survey," in *Advanced Information Networking and Applications Workshops, 2009. WAINA'09. International Conference on*. IEEE, 2009, pp. 636–641.

[3] Q. Jiang and D. Manivannan, "Routing protocols for sensor networks," in *Consumer Communications and Networking Conference, 2004. CCNC 2004. First IEEE*. IEEE, 2004, pp. 93–98.

[4] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: a survey," *IEEE wireless communications*, vol. 11, no. 6, pp. 6–28, 2004.

[5] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad hoc networks*, vol. 3, no. 3, pp. 325–349, 2005.

[6] A.-S. K. Pathan, H.-W. Lee, and C. S. Hong, "Security in wireless sensor networks: issues and challenges," in *Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference*, vol. 2. IEEE, 2006, pp. 6–pp.

[7] M. Teymourzadeh, R. Vahed, S. Alibeygi, and N. Dastanpor, "Security in wireless sensor networks: Issues and challenges," *International Journal of Computer Networks and Communication Security*, vol. 1, 2013.

[8] P. Bonnet, J. Gehrke, and P. Seshadri, "Towards sensor database systems," in *International Conference on Mobile Data Management*. Springer, 2001, pp. 3–14.

[9] R. Govindan, J. Hellerstein, W. Hong, S. Madden, M. Franklin, and S. Shenker, "The sensor network as a database," Citeseer, Tech. Rep., 2002.

[10] S. R. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "Tinydb: an acquisitional query processing system for sensor networks," *ACM Transactions on database systems (TODS)*, vol. 30, no. 1, pp. 122–173, 2005.

[11] S. Madden, J. Hellerstein, and W. Hong, "Tinydb: In-network query processing in tinyos," 2003.

[12] D.-O. Kim, L. Liu, I.-S. Shin, J.-J. Kim, and K.-J. Han, "Spatial tinydb: a spatial sensor database system for the usn environment," *International Journal of Distributed Sensor Networks*, 2013.

[13] P. Di Felice, M. Ianni, and L. Pomante, "A spatial extension of tinydb for wireless sensor networks," in *Computers and Communications, 2008. ISCC 2008. IEEE Symposium on*. IEEE, 2008, pp. 1076–1082.

[14] K. Park, B. Lee, and R. Elmasri, "Energy efficient spatial query processing in wireless sensor networks," in *Advanced Information Networking and Applications Workshops, 2007, AINAW'07. 21st International Conference on*, vol. 2. IEEE, 2007, pp. 719–724.

[15] Y. Yao and J. Gehrke, "The cougar approach to in-network query processing in sensor networks," *ACM Sigmod record*, vol. 31, no. 3, pp. 9–18, 2002.

[16] Y. Yao, J. Gehrke *et al.*, "Query processing in sensor networks." in *Cidr*, 2003, pp. 233–244.

[17] J. Gehrke and S. Madden, "Query processing in sensor networks," *IEEE Pervasive computing*, vol. 3, no. 1, pp. 46–55, 2004.

[18] E. Buchmann, N. Tatbul, and M. Nascimento, "Query processing in sensor networks," *Distributed and Parallel Databases*, vol. 29, no. 1, pp. 1–2, 2011.

[19] S.-J. Chen, D. C. Hovde, K. A. Peterson, and A. W. Marshall, "Fire detection using smoke and gas sensors," *Fire Safety Journal*, vol. 42, no. 8, pp. 507–515, 2007.

[20] J. Seebamrungsat, S. Praising, and P. Riyamongkol, "Fire detection in the buildings using image processing," in *Student Project Conference (ICT-ISPC), 2014 Third ICT International*. IEEE, 2014, pp. 95–98.

[21] W. Wang and H. Zhou, "Fire detection based on flame color and area," in *Computer Science and Automation Engineering (CSAE), 2012 IEEE International Conference on*, vol. 3. IEEE, 2012, pp. 222–226.

[22] J. Chen, Y. He, and J. Wang, "Multi-feature fusion based fast video flame detection," *Building and Environment*, vol. 45, no. 5, pp. 1113–1122, 2010.