

Increasing the Precision of Noise Source Detection System Using KNN Method

Parlin Nando^{*1}, Aji Gautama Putrada², Maman Abdurohman³

^{1,2,3}Universitas Telkom/School of Computing

parlinnando@alumni.telkomuniversity.ac.id^{*1}, ajigps@telkomuniversity.ac.id²,

abdurohman@telkomuniversity.ac.id³

Abstract

This paper proposes Accurate Noise Source Detection System using K-Nearest Neighbor (KNN) Method. Noise or sound intensity is usually measured in decibels (dB). In an educational environment the recommended noise index limit is 55 dB. It means that noise louder than that limit is prohibited. While a loud noise in a campus area occurred, it will be troublesome for the authorities to deal with the matter. This paper proposes a noise source detection system that can locate the position of the noise source. This system used Df analog V2 voice sensor for detecting the loud noise intensity. A microcontroller with WiFi capabilities will allow the system to communicate with an Internet of Things (IoT) platform that can perform a learning method to detect the location of the loud noise source. KNN method is used as the learning method. The result shows a user is able to get a warning related to the noise that occurs in an area at once. The precision position of the noise source can be detected with 70% average accuracy rate.

Keywords: Loud noise, noise source detection, Internet of Things, K-Nearest Neighbor, Wireless Sensor Network

1. Introduction

Noise is one of the things that must be avoided in the educational environment, for example the campus environment [1] [2] [3] [4] [5]. The campus environment that is free of noise is one of the important aspects that support a conducive campus environment condition. With a conducive condition in the campus environment, it will certainly make every student learn well.

Noise has a frequency unit defined in Hertz (Hz) and a unit of intensity expressed in decibel (dB). The noise level standard for the area or environment for educational activities is in accordance with the Minister of Environment Decree No. KEP-48 / MENLH / 11/1996 have a noisy threshold value between 45 to 55 dB.

A study [6] has been done before to address the same noise problem, but in this study only aims to detect if noise occurs in a room, without knowing exactly where the position is and the accuracy of the noise source.

From various problems related to noise, a WSN Performance Analysis was made with the KNN Method which is useful for detecting noise with precise accuracy. By using the KNN method, accuracy can be detected by binaural signals consisting of inter-aural time difference (ITD) and intermural level difference (ILD) [7]. Binaural signals are obtained by converting sound source signals and impulse binaural room impulse responses. This binaural signal contains head related transfer function (HRTF) and a room impulse response that represents acoustic conditions.

This research aims to design a system that can detect not only loud noise but also the location of the source of the loud noise within a campus environment using KNN methods. To analyze the performance of the system it will be tested in a limited test bed using Telkom University as a case study. This limited test bed will be a certain area in the complex of Telkom University. That certain area is a building block comprises of three different buildings and one empty field. The names of the buildings are Building E, Building F, and Building G. The map of the area will be provided further in this paper.

The system that will be a mixture of Data Analysis and Embedded System will be in a form of IoT Architecture. Where there will be communication between an IoT End Device and an IoT Platform. Acting as the IoT End Device is a Wireless Sensor Network comprising of three identical Sensor Nodes. Each Sensor Node will be placed in each Building in the Block mentioned earlier. Each node will comprise of a microcontroller with wireless communication capabilities and a

sound sensor that detects sound intensity. The WSN uses Star as the topology with one gateway node.

The IoT Platform will be Thingspeak, an open source IoT Platform that uses HTTP Rest as means of communication and has the function of embedding Matlab, which will be used to do the computing of the KNN Method. Nine different points will be used to train and test the KNN method [8].

After this introduction the paper will further discuss about related works, containing researches that are similar to this research and are used as bases and comparison. Then the paper will discuss about the design of the system, specifically about the hardware and software design of the IoT end device and the design of the KNN method. Further test results will be discussed. Test results show the performance analysis of this paper's proposed method. Finally a conclusion will be made in the last chapter.

2. Related Works

In general, a sound noise detection system is not a new thing. There have been many previous experiments that have attempted to make sound noise detection systems in different ways. One example is [9] which detects sound noise and breaks it into three types of noise; irritating noise, masking noise, and damaging noise. In this study a noise detection system will be made that can detect the type of noise and classify the types of noise that occur at a time. The type of noise that is detected and classified is steady state wide band noise, intermittent noise, and explosive noise. The purpose of this study is that noise that occurs in an environment can be distinguished and classified according to type.

In the process of making a sound noise detection prototype system, it will implement WSN. [10] explained that with the presence of many sensors that would later communicate with each other, the localization was important to do considering that the uncertainty in determining the exact location of the sensors used could have occurred. One example is in the humidity monitoring system in the forest, where there are thousands of sensors used

There are several localization methods that apply WSN in it, such as one example in applying the method [11] K-Nearest Neighbor (KNN). The scenario used in this paper is to limit that it will use a node that will be called a blind node that will communicate with similar nodes, where similar nodes can know and can tell where they are. Several other similar nodes are called anchor nodes or landmarks that know the exact position of their location.

In [12] proposing a sound source direction detection algorithm using the GMM method has become the standard estimate due to its better performance compared to other estimation methods, the direction of the sound source can be detected by binaural signals consisting of interaural time difference (ITD) and interaural level difference (ILD), many studies are based on the microphone arrays method which describes a method that uses several microphones to obtain high performance in sound direction detection. However, the number of microphones can be reduced to save research and maintenance costs by using K-Nearest Neighbor (KNN).

In 2015 research was conducted on sound monitoring that utilizes sound localization [13]. The novelty of this research is the application of Bluetooth Low Energy (BLE). The system proposed in this study was made to detect the level of noise pollution produced by development projects in residential areas and work areas. To conduct monitoring and detection of sound origin, this system uses three WSN nodes. In this study the method used for voice localization is time synchronization and time difference that utilizes triangulation from existing nodes.

A study on sound localization for indoor sound monitoring was carried out in 2015 [14]. This study states that indoor sound monitoring is done to control the sound in the room so as not to violate existing regulations. Workers are targeted for detecting the origin of sound in this study. Moving node detection in this study utilizes ultra-wide band (UWB) localization in a room measuring 30x10x3 m. The tracking is then enhanced by Bayesian filters. The results of this study suggest that the accuracy of detection on a straight track is different from the detection of a curved track.

A study to apply noise monitoring at the airport was carried out in 2017 [15]. This research utilizes Phased Microphone Array. Phased Microphone Arrays are used to filter aircraft sounds with other noises, from traffic to barking sounds. This study states that sound monitoring at the airport is important so that the surrounding population is not disturbed by noise caused by the airport. This study used a beam-forming technique called frequency domain beam-forming (FDBF). The Phased Microphone Array used in this study utilizes a 24-channel compact Optimav

microphone array. This array is equipped with 24 microphones with a multiarm spiral layout. The results of this study state that from several algorithms used for localization, TIDY algorithm is the best algorithm in aircraft localization for sound monitoring at airports using Phased Microphone Array.

A research on urban sound mapping using the Kriging method was carried out in 2018 [16]. This study states that conventional sound detection carried out spatially cannot produce sound mapping with good performance. Instead of using spatial voice mapping, this study proposes a sound mapping method in which sound data collection is done by walking through city streets. The road used in this study is the streets of Paris. Collection of voice data is done by walking several times through the same path. Sound mapping is then constructed based on a set of sound data using an interpolation method called Kriging. The results of this study are that this study succeeded in showing performance comparisons between ordinary Kriging and Universal Kriging in sound map interpolation.

A research on sound localization with the KNN method was conducted in 2018. This research is IoT themed, where researchers highlight the power of software, computing, and platforms to support the use of low-cost hardware and sensors. The KNN type used in this study is called Representative KNN (RKNN). RKNN chooses the optimum KNN model to implement sound localization. The tools involved in this research are one speaker and one Microphone IoT node. Localization training is used by measuring 16 different zones in a 4x4 two dimensional shape on a table with an approximate size of 1 m². The accuracy of this study in locating sound is 88.02%.

3. System Design

3.1 The Sound Source Detection System End Device

The following are system requirements, namely:

1. Perform loud noise simulation at several predetermined points to obtain training data
2. Test the noise at several points according to predetermined conditions to obtain testing data
3. Checking between training data and testing data using the K-Nearest Neighbor (KNN) algorithm to accurately determine the location of the noise source as well as calculating the accuracy of the system being tested and displaying a warning to the user
4. Use of MATLAB in the process of checking between training data and testing data using the K-Nearest Neighbor (KNN) algorithm

This Research analyzes the performance of a microcontroller-based noise detection system prototype. In making the prototype 3 sensors were used in accordance with each testing point, namely an Analog Robot V2 Df sensor to detect incoming sound in the circuit.

In addition to using 3 sensors, the system prototype also uses a NodeMCU device that is integrated with the ESP8266 WiFi Module. The variable value obtained from the sensor is used as input for the NodeMCU which is then processed. If the NodeMCU has obtained the variable value from the sensor, the value data is sent via wireless communication via ThingSpeak. Figure 1 shows the system environment.

The system prototype made is to calculate the intensity of the incoming sound using the Analog Robot V2 Df Sound Sensor. The system starts to do sound detection of points in building E, Building F, and Building G. By utilizing the ESP8266 WiFi module that is embedded in the NodeMCU chip. Data from the sensor is then forwarded to the server using ThingSpeak and the existing data will be stored. ThingSpeak functions to receive data from the sensor. Then MATLAB contained in ThingSpeak will display the warning which building is experiencing noise [17].

The system built in this research consists of three parts, namely: electronics, gateway and application. The electronic part consists of several devices namely Analog Sensor V2 Df Sound Sensor, and NodeMCU. The construction of the gateway uses the Wifi ESP8266 module which is integrated with the NodeMCU which will receive data from the electronic part and forward it to the server. The data sent is in the form of current data from the V2 Analog robot Df Sound sensor.

In this system series using 3 analog robot V2 Df sensors, each of which is a V2 robot Analog Df sensor connected to 3 NodeMCUs. Each data from the 3 sensors is divided into 3 channels at ThingSpeak (Building E, Building F, and Building G).

The use of these 3 channels helps in handling noise problems in the Telkom University area, the parable is that if a series of noise detection prototypes is added, the whole system does

not need to be changed but only adds a new channel to ThingSpeak according to the number of additional circuits in the Telkom University area.

The sensor's identity is known based on the differences in each channel in ThingSpeak. In this final project, three Analog V2 Df Sound Sensor are used, where each sensor data will send to ThingSpeak with different channels, as well as later if this system will add sensors in the future, only add the channel to ThingSpeak according to the number of sensors used. Each channel (Building E, Building F, Building G) will receive data from 3 sensors that have been published by NodeMCU through ThingSpeak by utilizing the ESP8266 WiFi module.

In addition to having 3 channels, there is also 1 additional channel that is used to store testing data from the three sensors in question when the actual test has been carried out. Data contained in the 3 channels will be processed in MATLAB using the K-Nearest Neighbor (KNN) algorithm to provide warning results related to the noise conditions that occur. Figure 2 provides an explanation of the flow of the whole process.

The scheme that applies to this detection system prototype is that at the beginning 3 Df sensors Analog V2 sensors will detect the sound intensity around the sensor, then each sensor sends data to ThingSpeak using the WiFi module. ThingSpeak will receive data from 3 sensors and will then be processed by MATLAB using the KNN method.

At this stage a system requirements analysis such as hardware and software is carried out that supports the execution of this final project. After the analysis process is carried out, the system specifications are as follows. Functional Requirements

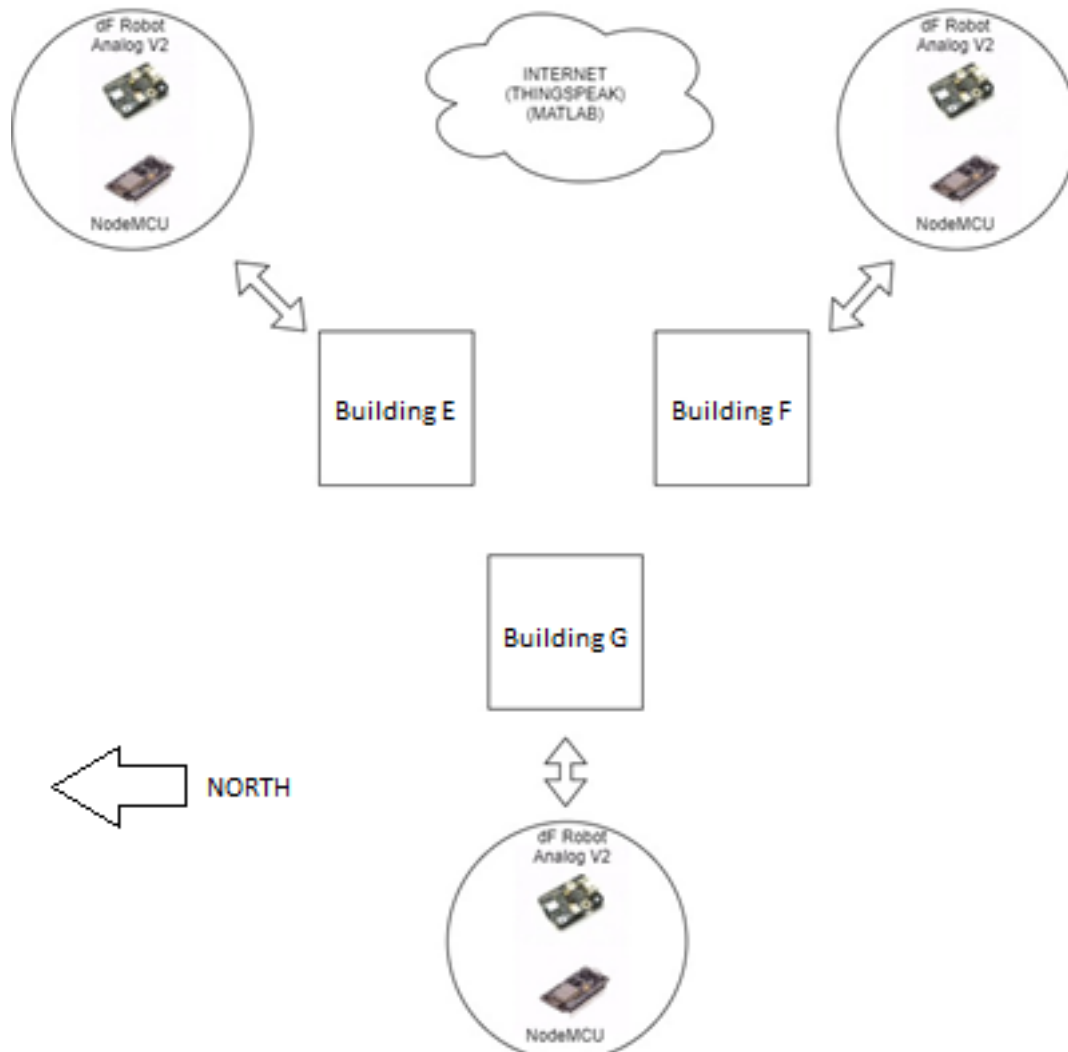


Figure 1. Loud Noise Source Detection System Diagram.

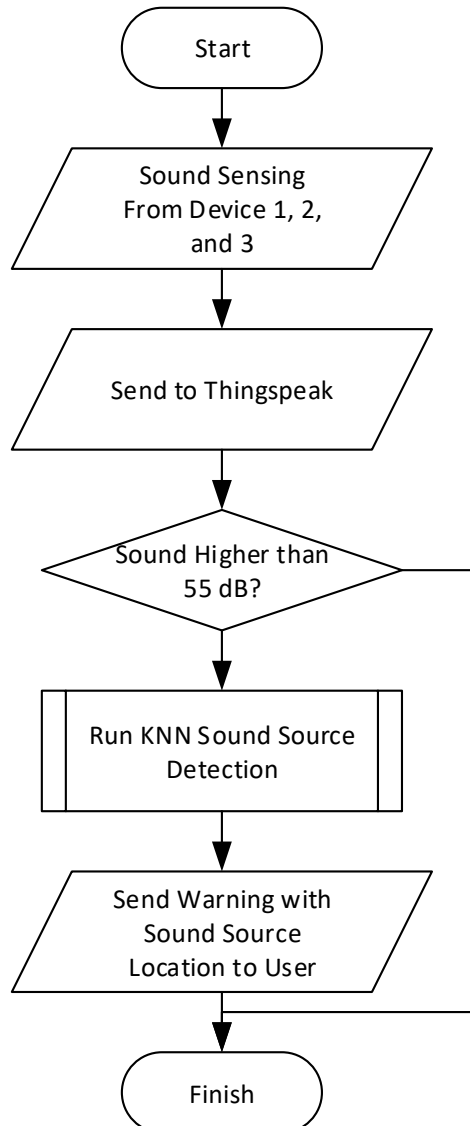


Figure 2. Flow Diagram of the Loud Noise Source Detection System

Functional Requirements of the system consist of:

1. Selecting the type of sensor based on existing references [5] which is suitable for noise detection;
2. Installation of analog V2 Sound Df sensor in the circuit;
3. The process of displaying the results of reading data from the sensor in the form of a line chart that can be updated in real-time;
4. The process of detecting noise from the circuit as well as the process displays a warning from MATLAB on ThingSpeak;

Performance Requirements

The Performance Requirements system are:

1. Df Sensor Analog V2 can detect sound;
2. Data from the Df Analog V2 Robot sensor can be sent to the server using ThingSpeak;
3. In accordance with the data received from the server, a warning will appear in the integrated MATLAB ThingSpeak;
 1. The first condition is the emergence of a warning "Noise in Point 1" when the Analog V2 Df Robot sensor exceeds above 55;
 2. The second condition is the emergence of the warning "Noise in Point 2" when the Df Robot Analog V2 sensor exceeds above 55 dB on channel 2;

3. The third condition is the emergence of a warning "Noise in Point 3" when the Analog V2 Df Robot sensor exceeds above 55;
4. The fourth condition is the emergence of a warning "Noise in Point 4" when the Analog V2 Df Robot sensor exceeds above 55;
5. The fifth condition is the emergence of a warning "Noise in Point 5" when the Analog V2 Df Robot sensor exceeds above 55;
6. The sixth condition is the emergence of a warning "Noise in Point 6" when the Analog V2 Df Robot sensor exceeds above 55;
7. The seventh condition is the emergence of a warning "Noise in Point 7" when the Analog V2 Df Robot sensor exceeds above 55;
8. The eighth condition is the emergence of a warning "Noise in Point 8" when the Analog V2 Df Robot sensor exceeds above 55;
9. The ninth condition is the emergence of a warning "Noise in Point 9" when the Analog V2 Df Robot sensor exceeds above 55;
4. In accordance with the data received from the server, the line chart will appear in ThingSpeak;

Experiments were carried out in the Telkom University area including in Building E, Building F, and the GSG Building. A noise detection system built concerning stipulation of noise levels that are permitted in schools and the like. The sensor specifications used are Df Robot Analog V2 with specifications of voltage 3.3 V to 5.0 V and analog interfaces.

3.2 System Specifications

The Hardware Specifications of IoT End Device, as shown in Table 1, are as follows:

1. Df sensor robot analog sensor V2. It is a sound capture sensor
 - o Voltage: 3.3 V to 5.0 V
 - o Interface: Analog
 - o Size: 22x32mm
2. NodeMCU. It is an electronic device used for microcontrollers to convert analog values from sensors to decibels.
 - o CPU: ESP8266
 - o Memory: 128kBytes
 - o Storage: 4Mbytes
 - o Power: USB

Software specifications of the entire system built are as follows:

- o Arduino 1.8.2 for uploading and running program code embedded in the nodeMCU
- o MATLAB application
- o The Google Chrome Web Browser that is used as a medium to access ThingSpeak

Table 1. Hardware System Specifications

Hardware	Hardware Type	System Hardware Functionalities
Microcontroller	NodeMCU	The system controller device and as the sensor data transmitting medium on ThingSpeak via WiFi.
Sound Sensor	DF Robot Analog V2	As a Sound Intensity detector that adjusts to the indication of Sound Intensity element.

3.3 KNN Method Design

This testing process begins by comparing the existing training data with the testing data obtained when conducting monitoring testing. From the existing testing data, the system will be able to determine which pattern or position is closest to the sensor, using the Euclidean Distance formula (Equation 1) [18].

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^n (a_r(x_i) - a_r(x_j))^2} \quad (1)$$

where $d(x_i, x_j)$ is Euclidean Distance, (x_i) is i^{th} record, (x_j) is j^{th} record, (a_r) is r^{th} record, i, j is $1, 2, 3, \dots, n$. The K in K -Nearest Neighbor method determines the amount of points used as reference to decide the class of the unclassified data. For example $K = 5$ means that which class are most mentioned out of 5 closest training data points to the unclassified point as calculated by the Euclidean Distance formula above. $K = 1$ is used in this research's proposed method.

4. Results and Discussion

4.1. Sensor Test Results and Scatter Diagram

Twenty loud noise data sensed from three identical sound sensors located in three different places were collected from each nine points of training location resulting in total 180 data.

Scatter plots can show distribution patterns and behavior [19]. These data were inserted into a three dimensional scatter diagram. The diagram can be seen in Figure 3. Each color representation shows a different point of location. It can be seen that the data is scattered and a mixture of point data occurs in the scatter diagram.

The design of the system circuit consists of several parts, namely the sensor part of the Analog Robot Df V2. This noise detection design uses 3 analog V2 Df Robot sensors coupled with NodeMCU. Figure 4.

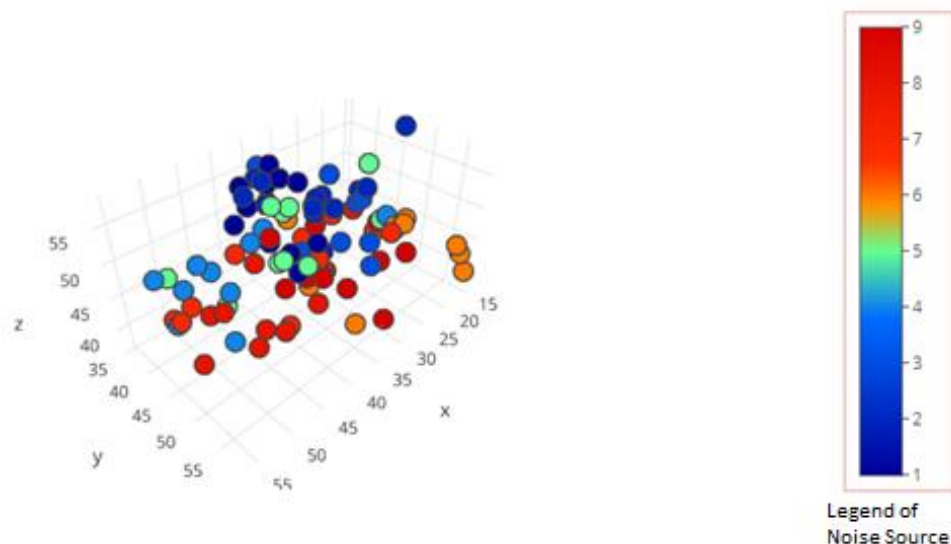


Figure 3. Scatter Chart of Sound Data coming from 9 Different Points.

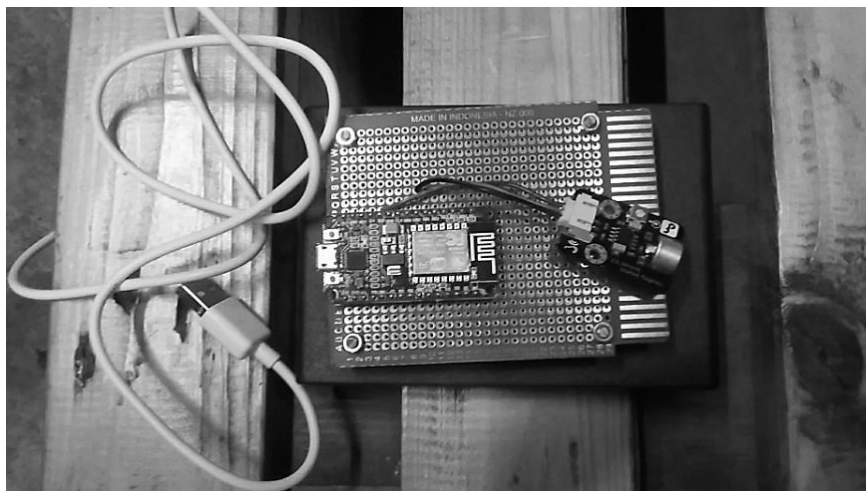


Figure 4. Each End Device Implementation comprises of a NodeMCU and a Sound Sensor (DF Robot Analog V2)

4.2. Functional Test Results

The scenarios carried out in this time testing are with 9 different locations, starting with making noise at 9 points alternately, Df Analog V2 Robot Sensor placed near each building and then detecting incoming sounds, then sending data to ThingSpeak.

Figure 5 shows the system implementation of hardware and software design, meanwhile, Figure 6 shows the test bed in the real world. The system detects loud sound source of nine predefined points from three identical sound sensors located in three different places, sends it to an IoT platform that has the capability to calculate the points location through KNN method. As seen in Figure 3 previously in succeeds to detect the sound intensity, however the next section will discuss about laboratory performance results of the KNN method as the sound source detection.

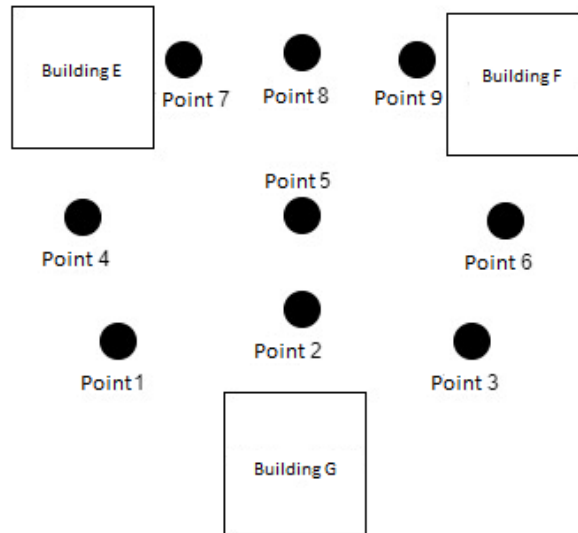


Figure 5. The Map of Each Sensor Placement and Each Testing Point Location



Figure 6. (Left) The actual test bed (an Area between Building E, Building F, and Building G). (Right) The deployment of the End Device at the actual test bed (Sensor Node of Building E)

4.3 Laboratory Performance Test of the KNN Method

As previously seen in Figure 3 the system has succeeded to detect a loud sound source through three identical sound sensors distributed in three different locations. The data collected at that section is used as training data for the KNN Method. The KNN Method implements formula (1) from section 3 above using $K = 1$. As many as 20 data each from 9 predefined location points

are collected to test the performance of the KNN method in determining loud noise source. The test is performed in a laboratory with the size of 4 m x 3 m. Table 2 shows the results of the test in forms of a confusion matrix.

Precision, Recall, and Accuracy will be the metric of the performance test of the KNN method. Precision and Recall are two metrics that are useful when dealing with imbalanced datasets [20]. Precision, Recall, and Accuracy are defined as the formula as follow Equation 2, 3, and 4.

$$Precision = \frac{TP}{FP + TP} \quad (2)$$

$$Recall = \frac{TP}{FN + TP} \quad (3)$$

$$Accuracy = \frac{TN + TP}{FN + TN + FP + TP} \quad (4)$$

Where FP is False Positive, TP is True Positive, FN is False Negative, and TN is True Negative.

The green highlighted cell in the confusion matrix in Table 2 is the True Positive (TP) of the test. Everything outside the green highlighted cell that is greater than 0 is either False Positive (FP) or False Negative (FN). Everything outside the green highlighted cell equal to 0 is a True Negative (TN).

An FP will result in low scores of Precision. It means that the system detects scores outside of the class as part of the class. An FN will result in low scores of Recall. It means that the system fails to detect all part of the class as part of the class.

The value shown in the green highlighted cell shows the majority of the result which is a good thing; however, the best recall rate is produced by point 4. That is 85%, meaning that at that point the system failed to detect 3 out of 20 actual loud noise source locations.

Table 2. Confusion Matrix of the Results of KNN Classification Process on Predicting Sound Source

	predicted									Total	Recall	
	1	2	3	4	5	6	7	8	9			
Point	1	2	3	4	5	6	7	8	9			
actual	1	13	3	0	3	1	0	0	0	0	20	65.0%
	2	3	13	4	0	0	0	0	0	0	20	65.0%
	3	3	2	10	3	2	0	0	0	0	20	50.0%
	4	0	0	1	17	2	0	0	0	0	20	85.0%
	5	0	0	0	3	15	2	0	0	0	20	75.0%
	6	0	0	0	0	1	15	3	1	0	20	75.0%
	7	0	0	0	0	0	1	15	4	0	20	75.0%
	8	0	0	0	0	0	0	3	15	2	20	75.0%
	9	0	0	0	0	0	0	2	5	13	20	65.0%
Total	19	18	15	26	21	18	23	25	15	180		
Precision (%)	68.4	72.2	66.7	65.4	71.4	83.3	65.2	60.0	86.7		Accuracy=70.0%	

The best precision rate is produced by point 6. That is 83.3%, meaning that at that point the system detected three loud noise source locations that were actually not from point 6.

Based on the result of experiment, it can be concluded the KNN method is not optimal enough to precisely and accurately determine the location of a loud noise source and further method developments are in need to increase the performance.

4.4 Actual Test Bed Performance Test

In this section, the test is to put the end devices between the buildings which are the actual test bed. Three end devices each deployed at Building E, Building F, and Building G. Between these buildings is an empty field. Figure 6 shows the condition of the location and the deployment of the end device.

Each end device connects directly to Thingspeak via internet using HTTP protocol. These devices send real time data to the IoT platform. The data is in form of sound intensity. The metric is dB (decibel).

It can be seen in Figure 6 that in the empty field there are children playing football. As they play, these children emit loud noise. This is a good moment to test the method in detecting loud noise source.

The results of the above tests obtained the results of the warning "Noisy Point 8" where the warning was in accordance with the training data that had been made before as seen in Figure 5. Concluding that in actual test bed, the system can accurately determine the location of a loud noise source in campus area.

4. Conclusion

Based on several experiments in laboratory as well as in the real world, it can be concluded that the implementation of noise source detection systems can effectively be implemented using the NodeMCU microcontroller, Df Analog Robot V2 Sensor, and Thingspeak. The results show the performance of system using KNN method has an average accuracy rate of 70.56%. When testing in the test bed, which is a limited environment within the Telkom University campus area, the result of the test produces a warning that matches the training data. There are opportunities for improving the system such as implementing a self-developed IoT platform, using a python server for computing the KNN method, implement others algorithms as a comparison to KNN, and expand the number of sensors in other buildings.

References

- [1] P. H. T. Zannin, M. S. Engel, P. E. K. Fiedler, and F. Bunn, "Characterization of environmental noise based on noise measurements, noise mapping and interviews: A case study at a university campus in Brazil," *Cities*, Vol. 31, Pp. 317–327, 2013.
- [2] D. Çolakkadioğlu, M. Yücel, B. Kahveci, and Ö. Aydınol, "Determination of noise pollution on university campuses: a case study at Çukurova University campus in Turkey," *Environ. Monit. Assess.*, Vol. 190, No. 4, Pp. 203, 2018.
- [3] J. S. Sudarsan and S. Nithiyantham, "Study on Noise Pollution at SRM University Campus (South India–Asia)," *Adv. Sci. Eng. Med.*, Vol. 8, No. 11, Pp. 896–905, 2016.
- [4] D. Thattai, J. S. Sudarsan, R. Sathyanathan, and V. Ramasamy, "Analysis of noise pollution level in a University campus in South India," in *IOP Conference Series: Earth and Environmental Science*, 2017, Vol. 80, Pp. 012053.
- [5] N. M. S. Purwaningsih *et al.*, "Analysis of Noise Pollution: A Case Study of Malaysia's University," *J. Int. Dent. Med. Res.*, Vol. 11, No. 1, Pp. 330–333, 2018.
- [6] B. N. Indratama, M. Abdurrohman, and S. Prabowo, "Sistem Monitoring Kebisingan Lingkungan Pendidikan berbasis OpenMTC," in *Seminar Nasional Komputer dan Informatika*, 2018.
- [7] J. Davila-Chacon, S. Magg, J. Liu, and S. Wermter, "Neural and statistical processing of spatial cues for sound source localisation," in *The 2013 International Joint Conference on Neural Networks (IJCNN)*, 2013, Pp. 1–8.
- [8] J. A. Waller, S. L. Dance, and N. K. Nichols, "On diagnosing observation-error statistics with local ensemble data assimilation," *Q. J. R. Meteorol. Soc.*, Vol. 143, No. 708, Pp. 2677–2686, 2017.
- [9] Y. Zhang, S. Xing, Y. Zhu, F. Yan, and L. Shen, "RSS-based localization in WSNs using Gaussian mixture model via semidefinite relaxation," *IEEE Commun. Lett.*, Vol. 21, No. 6, Pp. 1329–1332, 2017.

- [10] L. Sun and Q. Cheng, "Indoor sound source localization and number estimation using infinite Gaussian mixture models," in *2014 48th Asilomar Conference on Signals, Systems and Computers*, 2014, Pp. 1189–1193.
- [11] L. Pan and J. Li, "K-nearest neighbor based missing data estimation algorithm in wireless sensor networks," *Wirel. Sens. Netw.*, Vol. 2, No. 02, Pp. 115, 2010.
- [12] N. A. Dieng, M. Charbit, C. Chaudet, L. Toutain, and T. B. Meriem, "Indoor localization in wireless networks based on a two-modes gaussian mixture model," in *2013 IEEE 78th Vehicular Technology Conference (VTC Fall)*, 2013, Pp. 1–5.
- [13] J. Hughes, J. Yan, and K. Soga, "Development of wireless sensor network using bluetooth low energy (BLE) for construction noise monitoring," *Int. J. Smart Sens. Intell. Syst.*, Vol. 8, No. 2, Pp. 1379–1405, 2015.
- [14] N. Prodi, A. Conti, F. Lodi, S. Bartoletti, and J. S. Dekhordi, "Localization-assisted indoor acoustical data monitoring," 2015.
- [15] K. Chelliah, G. Raman, M. R. Aldeman, and H. J. Patel, "Demonstration of the possibilities of using a phased microphone array as the next-generation airport noise monitoring system," *Transp. Res. Rec.*, Vol. 2600, No. 1, Pp. 20–26, 2016.
- [16] P. Aumond *et al.*, "Kriging-based spatial interpolation from measurements for sound level mapping in urban areas," *J. Acoust. Soc. Am.*, Vol. 143, No. 5, Pp. 2847–2857, 2018.
- [17] S. Pasha, "ThingSpeak based sensing and monitoring system for IoT with Matlab Analysis," *Int. J. New Technol. Res.*, Vol. 2, No. 6, 2016.
- [18] P.-E. Danielsson, "Euclidean distance mapping," *Comput. Graph. Image Process.*, Vol. 14, No. 3, Pp. 227–248, 1980.
- [19] P. E. Touchette, R. F. MacDonald, and S. N. Langer, "A scatter plot for identifying stimulus control of problem behavior," *J. Appl. Behav. Anal.*, vol. 18, no. 4, pp. 343–351, 1985.
- [20] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets," *PLoS One*, Vol. 10, No. 3, Pp. e0118432, 2015.

