



INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv



A Real-Time Application for Road Conditions Detection based on the Internet of Things

Lanny Sitanayah^{a,*}, Apriandy Angdresey^a, Evander Kristalino^a

^a Department of Informatics Engineering, Universitas Katolik De La Salle, Manado, Indonesia

Corresponding author: *lsitanayah@unikadelasalle.ac.id

Abstract— Bad road conditions may lead to road accidents, especially when drivers are unaware of potholes. The presence of potholes can increase from time to time and may get worse due to road age and bad weather. With the Internet of Things technology, vehicles on the road can be a means of collecting road condition data, such as vibration. The raw vibration data are useful only after they are processed into meaningful information. Information about the condition of roads can help other road users be aware of potholes. This paper proposes an Internet of Things application for road conditions detection. We design and implement a device comprising one NodeMCU ESP8266, one accelerometer gyroscope sensor to detect the existence of potholes based on the amount of detected vibration, and a GPS module to get the information about potholes' locations. For the web service, we use REST API so that users can get real-time potholes' information in the Android application. To cluster potholes based on detected vibration, i.e., deep, medium, and shallow, we implement the k -means clustering algorithm with $k=3$. The Android application utilizes a Google map to visualize potholes' locations and the result of clustering on a road map. We use colored pins to indicate the depth of potholes. Deep potholes are shown on the map using red pins, medium potholes using orange pins, and shallow potholes using green pins.

Keywords—Android application; sensor device; k -means clustering algorithm; data mining; potholes.

Manuscript received 22 Apr. 2022; revised 8 Jun. 2022; accepted 27 Jul. 2022. Date of publication 30 Sep. 2022. International Journal on Informatics Visualization is licensed under a Creative Commons Attribution-Share Alike 4.0 International License.



I. INTRODUCTION

Transportation is a very important supporting factor that influences the economic growth rate in modern society. Hence, it can be a driving force for the dynamics of the urban development of a city. In big cities, the development of transportation increases rapidly. This is due to the high mobility and activity of the urban population. Manado is one of the largest cities in Sulawesi Island. It is the capital of North Sulawesi province with an area of 162.53 km². Manado has 11 sub-districts with a population of around 451,916 people in 2020 [1]. Large cities like Manado have several types of roads, such as main, primary, secondary, and local.

Roads are important infrastructures for land transportation used by the community in their daily activities. In addition, roads are also important to accelerate the economic relationship between one region and the others. Because of these reasons, roads must be adequate to be used. Proper road conditions will support better mobility, provide comfort and ensure the safety of people on the roads. However, the higher the rate of daily activities, the higher the road traffic load. Road conditions may also change gradually because of road

age and bad weather. These cause damage to roads, such as potholes. Potholes reduce driving comfort and cause various problems, such as congestion, vehicle damage, and even traffic accidents. These problems may occur as road users do not go through these roads frequently, so they are not familiar and aware about the road conditions.

Nowadays, wireless-based network access and resources are developing and replacing the use of wired networks. The Internet of Things (IoT) is a current technology that is recently being developed because of its advantages in terms of functionality, performance support, and wireless capability. The Internet of Things aims to make it easier for people to interact with all devices connected to the Internet.

In digital age, plenty of data are generated every second from many sources. Inexpensive storage technology has made it possible to maintain all of the data. However, raw data are useless until they are processed to produce useful information. Therefore, it is necessary to analyze the raw data to extract some knowledge. Data mining provides an efficient way to analyze very large data sets and to extract some useful or possibly unexpected patterns in the data. In data mining, clustering is a technique to group the number of data or

objects into clusters or groups to find data with the same and different characteristics from objects in other clusters and calculate the optimal number of clusters to produce better clusters.

This paper proposes a design and implementation of a real-time application to detect road conditions by utilizing the Internet of Things technology. The contributions of this paper are a wireless system to detect road conditions and a clustering model to cluster potholes on the road based on their depths. The wireless system uses a sensing device that has one NodeMCU ESP8266, and a GPS module (Ublox NEO-6M-0-001 GY-GPS6MV2) to get the locations of potholes and also an accelerometer and gyroscope sensor (MPU-92/65) to detect vibration. Furthermore, the k -means clustering algorithm is utilized to cluster the depth of potholes as deep, medium, and shallow by using the detected vibration data. The clustering results are visualized on a road map in an Android application. Users can then use their smartphones to view real-time information regarding road conditions.

II. MATERIALS AND METHOD

The Internet of Things enables objects to connect by using the Internet and sending data and information over a network. With relatively affordable computers and ubiquitous wireless networks, it is possible to change anything from the smallest to the biggest. The Internet of Things is a result of the optimization of several devices to make it easier for humans to interact with all equipment connected to the Internet [2].

The Internet of Things has developed very rapidly in the last few years. This seamless communication can be done with every person, process, and object. All devices with or without batteries [3] – can share and collect data with low-cost computing. In today's interconnected world, digital systems can monitor, record, and adjust all interactions between connected things [4]. Consequently, humans as users are less involved in the implementation process.

Nowadays, the rapid development of information technology has produced piles of data. Data mining techniques can be applied to look at hidden or previously unknown knowledge from a data set. Several applications of data mining in the literature include using one of the data mining techniques to assist companies in classifying customers accurately based on customer asset outlier data [5], predicting the sex of tarantulas using a classification model [6], finding frequently connected or popular groups of users with common interests and recommending friends for mining a large social network [7], and analyzing public sentiment from tweets against the presidential candidates [8].

Applications of the Internet of Things and data mining techniques have been implemented [9], [10] to detect incoming and outgoing vehicles and count how many vehicles are entering and leaving a parking space. In this application, moving objects are classified as cars, motorcycles, or people. The real-time information provided by this Internet of Things application can supply users with complete information about the parking lot. Angdresey et al. [11] classified the sensor data from a swimming pool to determine its water quality. In this application, swimming pool water is classified as either clean or dirty. An application to monitor ornamental plant soil in a pot to recommend treatment for the plant [12]. In this application, three treatment categories were classified for

plants: need fertilizer, need water, or normal. These are based on sensor data, including soil pH, air temperature, soil moisture, and air humidity. An application to monitor water conditions in an aquarium has been proposed by Angdresey et al. [13]. This application will notify users about the water condition, whether it is normal or needs to be changed, based on water level, turbidity and temperature.

In data mining, different techniques are used – one of which is clustering. Clustering is a technique to identify object classes that have similarities. Using a clustering technique, one can determine the density and distance of area in an object space and identify the overall distribution pattern and correlation between attributes. The k -means clustering is a non-hierarchical clustering algorithm that works by partitioning data into several clusters or groups [14]. Hence, one cluster has data with the same characteristics, while two separate clusters have data with different characteristics.

Fakhi et al. [15] implemented k -means by dividing the problem into sub-problems that are handled independently on streaming multiprocessors from one or more GPUs, using the latest generation of GPUs with Compute Unified Device Architecture (CUDA). Daoudi et al. [16] compared the three most efficient implementations of the k -means algorithm. This study shows good results in acceleration effects for data sets. An adaptive image segmentation technique that uses the k -means algorithm to examine different image objects has been proposed by Venkatachalam et al. [17]. The aim is to produce accurate results with an easy process and to avoid bilateral input of k values. Grouping using the k -means algorithm gives good results.

The problem of detecting road conditions has been studied in the literature. A road condition detection system that consists of an Arduino and a smartphone has been proposed by Chen et al. [18]. In this system, sensor data from an accelerometer are evaluated in Arduino to find the acceleration's average and slop. Arduino then sends the evaluation results and raw data to the smartphone via Bluetooth. This system does not incorporate any data mining techniques for data analysis. A threshold was only used to determine the road conditions. If the vibration level is higher than the pre-determined threshold, the smartphone will automatically record the GPS position.

Road roughness classification is studied by Chen et al. [19], that used a GPS and an accelerometer to gather road roughness data. The data, including the three-axis acceleration, velocity, location, and time, are sent to a server for further processing. The server classifies the roughness levels, estimate the international roughness index, and analyzes the power spectral density of surface roughness. However, after the server analyzes the data, potholes' locations are not informed to road users. Meanwhile, participatory sensing to detect potholes has been studied by Medins et al. [20], that used Android OS-based smartphones with accelerometers as the hardware platform. They use several algorithms based on threshold and standard deviation to detect potholes.

In this paper, we use the Internet of Things concept together with a data mining clustering algorithm, i.e., the k -means algorithm, to detect potholes, cluster them based on their depth (deep, medium, and shallow), and inform users about their locations. We choose $k = 3$ [21]; potholes are

categorized into three damage levels based on their depths, i.e., lightly damaged, moderately damaged, and badly damaged. We firstly describe the wireless system as a whole and then describe the hardware and software separately. Specifically, we explain their detailed design and implementation.

A. The System

The wireless system to detect road conditions is illustrated in Figure 1. The sensor device in this system consists of one NodeMCU ESP8266, a GPS module (Ublox NEO-6M-0-001 GY-GPS6MV2) to detect the location, and an accelerometer gyroscope sensor (MPU-92/65) to detect vibration. As the web service to bridge end users and the sensor device, we utilize REST API (Representational State Transfer Application Programming Interface). Users can then use our Android application, which is installed on their smartphones, to access information regarding road conditions.

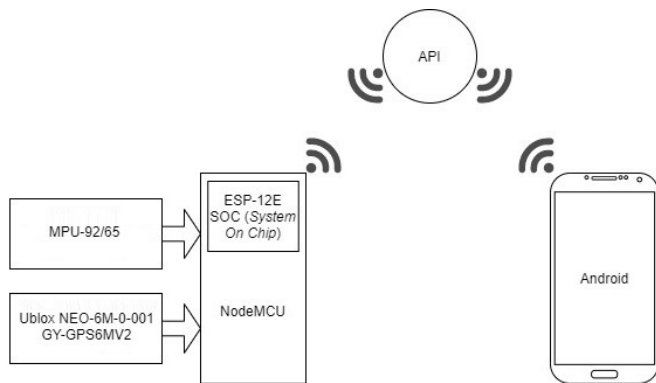


Fig. 1 The wireless system

A pothole is detected when one vibration data is below or above a threshold. When the sensor device detects a pothole, NodeMCU sends a request to API. Communication between NodeMCU and API is established by using the HTTP (Hypertext Transfer Protocol) methods. Data transmitted by the device are the latitude and longitude of the pothole's location and the three-axis accelerations. API uses the k -means clustering algorithm with $k = 3$ to process data stored in a database. Data are clustered to define the depth of potholes, i.e., deep, medium, and shallow. The clustering results are then recorded back to the database, organized in a JSON (JavaScript Object Notation) format, and then transmitted to the Android application by API.

B. The Hardware

The sensor device (hardware) design is depicted in Figure 2, where a NodeMCU ESP8266, a GPS module, and an accelerometer gyroscope sensor are connected on a breadboard. We connect the accelerometer's SCL pin to the D1 pin on NodeMCU, the accelerometer's SDA pin to D2 pin on NodeMCU, and the GPS's TX pin to D3 pin on NodeMCU, and the GPS's RX pin to D4 pin on NodeMCU. Then, we connect VCC pins on GPS and accelerometer to both 3V pins on NodeMCU. Furthermore, we connect GND pins on both GPS and accelerometer and the battery's negative pole to G pins on NodeMCU, while the battery's positive pole to VIN on NodeMCU. We show the summary of pin configuration

for the sensor device in Table 1 and the sensor device's real image in Figure 3.

TABLE I
PIN CONFIGURATION

NodeMCU	Accelerometer	GPS	Battery
D1	SCL	-	-
D2	SDA	-	-
D3	-	TX	-
D4	-	RX	-
3V	VCC	VCC	-
G	GND	GND	Negative pole
VIN	-	-	Positive pole

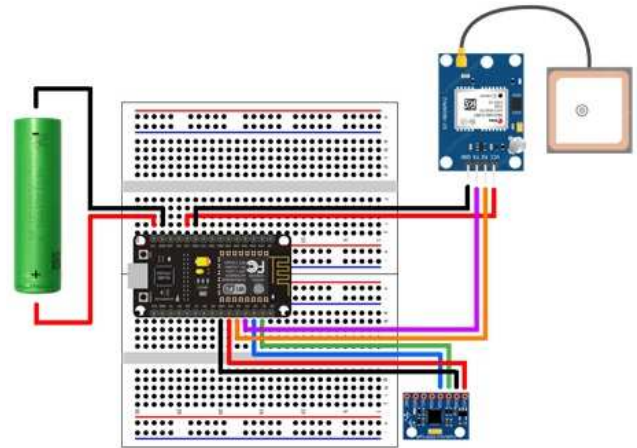


Fig. 2 Hardware design

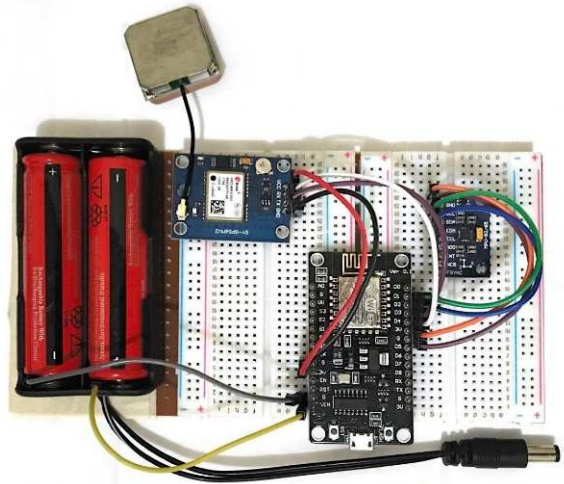


Fig. 3 The sensor device

C. The Software

The software is divided into programs on the sensor device (the client-side) and the server-side. Figure 4 shows a flowchart for the client-side, where the processes start from sensor initialization, data gathering, to data transfer. To minimize overhead, sensor data are sent only if potholes are detected. A pothole exists when an accelerometer reading is below or above certain thresholds. Processes on the client side are going on and on in a loop as long as the sensor device's battery still has power.

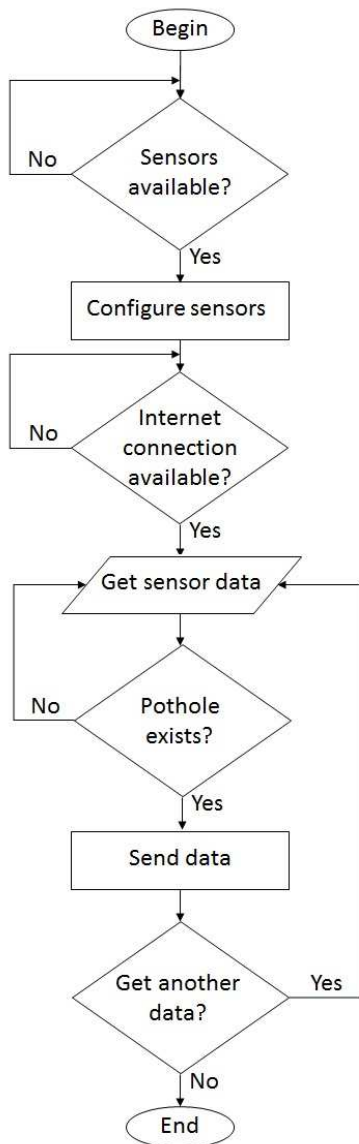


Fig. 4 Client flowchart

Figure 5 shows a flowchart for the server-side, where the workflow starts when the server receives data from the sensor device. The data are then stored and processed using the k -means clustering algorithm with $k = 3$. These include calculating centroids, calculating Euclidean distances to the centroids, and grouping data into clusters. The clustering results are sent to the Android application, and the processes repeat. We explain the k -means clustering algorithm in detail in the following section.

In this application, we implement three program modules: the k -means function, Google Maps API, and the HTTP methods. We use the k -means function with $k = 3$ to cluster vibration data into deep, medium, and shallow potholes. We use HTTP to connect the application, sensor device, API, and database. Finally, Google Maps API is used to inform users regarding potholes' locations on top of road maps.

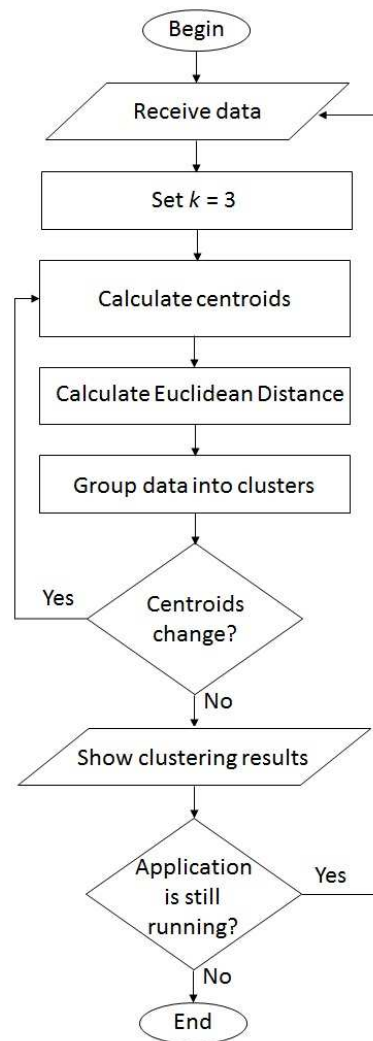


Fig. 5 Server flowchart

D. Object Clustering Method

Clustering is a data mining technique to organize data into several clusters or groups, where data between clusters have a minimum similarity and data in one cluster have a maximum similarity. Objects in a cluster are different from other clusters, but they have similar characteristics to each other in their clusters. In other words, clustering is a data segmentation that divides multiple data sets into groups according to their similarities. Two clustering methods can be used to group data: hierarchical clustering and partitioning. This paper uses the partitioning method, i.e., the k -means clustering algorithm.

The k -means clustering algorithm aims to minimize the objective function set during the clustering process by maximizing variations with data in other clusters and minimizing variations between data in a cluster. In the k -means clustering algorithm, firstly, we need to determine how many clusters we need, and the number of formed clusters must be less than the number of existing data ($k < n$). Secondly, we need to initialize a centroid and calculate distance space or data distance to the centroid by using the formula to calculate Euclidean distance as given in Equation 1, where D is the Euclidean distance from data to the centroid, a is data, and b is the centroid.

$$D(a, b) = \sqrt{\sum_{i=1}^n (b_i - a_i)^2} \quad (1)$$

Suppose we have four data with two variables as follows: $data_1 = \{1, 1\}$, $data_2 = \{2, 1\}$, $data_3 = \{4, 3\}$, and $data_4 = \{5, 4\}$. The initialization of centroids can be determined manually or randomly. The initial centroids of that data set are $C_1 = \{1, 1\}$ and $C_2 = \{2, 1\}$. For the following iterations ($iteration = 1, 2, 3, \dots, n$), new centroids are calculated by averaging the value of data in clusters. If new centroids are different from the previous ones, the process continues. However, if newly calculated centroids are the same as the previous ones, the clustering process is complete.

At $iteration = 1$, the data distances to centroid C_1 are as follows:

$$\begin{aligned} D(data_1, C_1) &= \sqrt{(1-1)^2 + (1-1)^2} = 0, \\ D(data_2, C_1) &= \sqrt{(1-2)^2 + (1-1)^2} = 1, \\ D(data_3, C_1) &= \sqrt{(1-4)^2 + (1-3)^2} = 3.605, \text{ and} \\ D(data_4, C_1) &= \sqrt{(1-5)^2 + (1-4)^2} = 5. \end{aligned}$$

Furthermore, the data distances to centroid C_2 are:

$$\begin{aligned} D(data_1, C_2) &= \sqrt{(2-1)^2 + (1-1)^2} = 1, \\ D(data_2, C_2) &= \sqrt{(2-2)^2 + (1-1)^2} = 0, \\ D(data_3, C_2) &= \sqrt{(2-4)^2 + (1-3)^2} = 2.828, \text{ and} \\ D(data_4, C_2) &= \sqrt{(2-5)^2 + (1-4)^2} = 4.243. \end{aligned}$$

Data are then grouped into clusters based on their shortest distances to centroids. For example, $data_1$ joins cluster 1 because $D(data_1, C_1) < D(data_1, C_2)$, while $data_2, data_3$, and $data_4$ join cluster 2 because $D(data_n, C_2) < D(data_n, C_1)$ for $n = 2, 3, 4$. After all data get their cluster labels, a new centroid for each cluster is calculated by averaging the value of its data. That is, by summing all value of data in that cluster and dividing it by the total number of data in that cluster. In this paper, we use $k = 3$ ac to the depth of potholes, *i.e.* deep, medium, and shallow. The Euclidean distance formula for $k = 3$ is shown in Equation 2.

$$D(X_i, C_i) = \sqrt{(a_i - C_i)^2 + (b_i - C_i)^2 + (c_i - C_i)^2} \quad (2)$$

For the following iterations, new centroids are calculated again. Data are grouped again into clusters according to the shortest distances to the previously found centroids. Then, the data value in each new cluster is averaged to find its new centroid. After obtaining cluster labels for all data and if all centroids do not change, the clustering process is complete.

III. RESULTS AND DISCUSSION

We conduct some experiments to evaluate the wireless system to detect road conditions in Manado – the capital of North Sulawesi province. We secure one sensor device on a motorcycle (at the front part). A user with the device attached to his motorcycle can acquire sensor data and update the database. On the other hand, a user without the device can only use the Android application to view the road conditions. In this experiment, our smartphone has Android 6.0 API 23 (Marshmallow).

The sensor device detects a pothole if one of the accelerometer readings is below or above certain thresholds. Based on some trials, we define the thresholds, *i.e.* a pothole is detected if a value from any accelerometer axes is greater than 5000 or less than -5000. When the sensor device detects

a pothole, it transmits the sensor data to the database through API.

Figure 6 shows data received by API from the sensor device when we check data transmission. The data include the three-axis accelerations denoted by ax , ay , and az , and also the coordinates of the pothole's location (latitude and longitude). The data set of potholes that are obtained by the sensor device is shown in Table 2. We cluster the data by using the k -means clustering algorithm with $k = 3$. Hence, the sensor data are clustered into shallow, medium, and deep potholes. The clustered sensor data are shown in Figure 7. The three axes represent the acceleration in the x , y , and z directions. Red dots are data clustered as deep potholes, orange dots are medium potholes, and green dots are shallow potholes.

```
Server Running on Port 2000
{ ax: '5755',
  ay: '4181',
  az: '-4199',
  latitude: '1.537704',
  longitude: '124.913127' }
```

Fig. 6 Data received by API

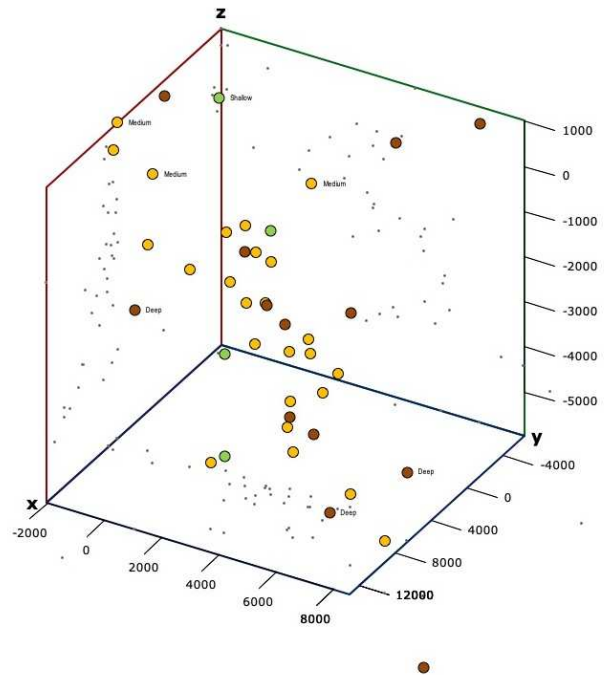


Fig. 7 Clustered sensor data

TABLE II
DATA SET OF POTHOLE

No	ax	ay	az	Latitude	Longitude
1	13077	1062	-1109	1.499638	124.870749
2	6549	-361	456	1.499335	124.87061
3	-5474	-1814	-303	1.499222	124.869891
4	5209	4776	979	1.499111	124.869906
5	5214	-2021	1026	1.499009	124.869917
6	6093	5678	-5982	1.498757	124.869938
7	6614	8548	-4438	1.497083	124.87017
...

20	11246	10575	-7554	1.522367	124.882357
21	5037	6092	-1669	1.519656	124.886146
22	6200	5055	-2561	1.516331	124.888372
23	7056	5736	-3146	1.511219	124.890596
24	6400	-1774	689	1.509261	124.890991
25	-5672	4314	-165	1.507727	124.89104
26	10401	7773	-4397	1.504435	124.891036
27	5622	3451	-230	1.50253	124.890676
...
40	5466	3291	-1942	1.513118	124.865191
41	11012	3083	-4482	1.514872	124.865073
42	11007	3539	-4256	1.517133	124.863902

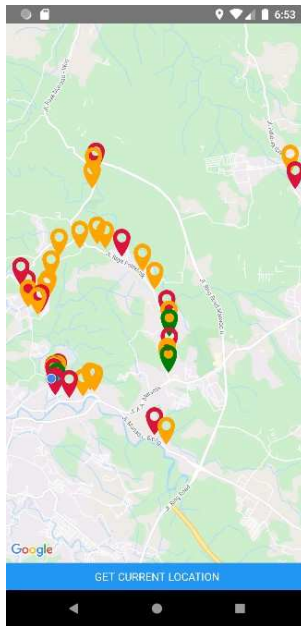


Fig. 8 Google map in the Android application

Figure 8 shows Google Maps in the Android application. It visualizes the result of clustering and the information about potholes' locations on a road map. In this application, we use colored pins to indicate the depth of potholes, where red pins indicate deep potholes, orange pins represent medium potholes, and green pins represent shallow potholes. In addition, we also provide pothole information, including the clustering result, latitude, and longitude in a text format. Figure 9 shows an example of a pothole description that is displayed when a user chooses a pin on the map.

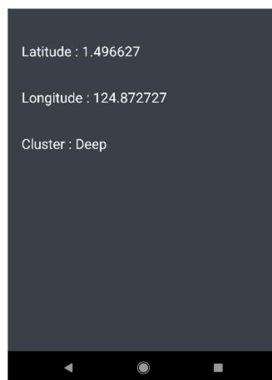


Fig. 9 A pothole description

IV. CONCLUSION

In this paper, we design and implement a real-time Internet of Things application for detecting road conditions. Our sensor device is built with a NodeMCU ESP8266, an accelerometer, and gyroscope sensor (MPU-92/65) to detect vibration, and a GPS NEO-6M module to obtain potholes' locations. As the web service, REST API is utilized to bridge end users and the sensor device. We implement the k -means clustering algorithm with $k = 3$ to cluster the vibration data to identify the depth of potholes, which are either deep, medium, or shallow. Users can view the visual representation of potholes' information on a road map in the Android application.

REFERENCES

- [1] "BPS Sulut (Badan Pusat Statistik Provinsi Sulawesi Utara)." <https://sulut.bps.go.id/indicator/12/45/1/jumlah-penduduk-kabupaten-kota.html> (accessed 2 February, 2021).
- [2] S. Narang, T. Nalwa, T. Choudhury, and N. Kashyap, "An Efficient Method for Security Measurement in Internet of Things," in *2018 International Conference on Communication, Computing and Internet of Things (IC3IoT)*, 2018, pp. 319-323.
- [3] J. Hester *et al.*, "Persistent Clocks for Batteryless Sensing Devices," *ACM Transactions on Embedded Computing Systems*, vol. 15, pp. 1-28, Aug. 2016.
- [4] C. Song, S. Liu, G. Han, P. Zeng, H. Yu, and Q. Zheng, "Edge Intelligence Based Condition Monitoring of Beam Pumping Units under Heavy Noise in the Industrial Internet of Things for Industry 4.0," *IEEE Internet of Things Journal*, 2022.
- [5] L. Zuo and J. Guo, "Customer Classification of Discrete Data Concerning Customer Assets Based on Data Mining," in *International Conference on Intelligent Transportation, Big Data Smart City (ICITBS)*, 2019, pp. 352-355.
- [6] A. Angdresey and M. Wongkar, "Identification of The Reproductive Apparatus of Tarantula Genus *Brachypelma* Using Linear Discriminant Analysis Method," in *International Conference on Electrical Engineering and Computer Science (ICECOS)*, 2018, pp. 409-414.
- [7] C. Fan, H. Hao, C. K. Leung, L. Y. Sun, and J. Tran, "Social Network Mining for Recommendation of Friends Based on Music Interests," in *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2018, pp. 833-840.
- [8] M. Wongkar and A. Angdresey, "Sentiment Analysis Using Naive Bayes Algorithm of the Data Crawler: Twitter," in *4th International Conference on Informatics and Computing (ICIC)*, 2019, pp. 1-5.
- [9] L. Sitanayah, A. Angdresey, and J. W. Utama, "A Low-Cost Vehicle Counting System based on the Internet of Things," *EPI International Journal of Engineering*, vol. 4, no. 1, pp. 14-20, 2021.
- [10] L. Sitanayah, A. Angdresey, and J. W. Utama, "An Internet of Things-Based Vehicle Counting System," in *2021 International Conference on Embedded Wireless Systems and Networks (EWSN)*, 2021, pp.175-176.
- [11] A. Angdresey, L. Sitanayah, and V. J. A. Sampul, "Monitoring and Predicting Water Quality in Swimming Pools," *EPI International Journal of Engineering*, vol. 3, no. 2, pp. 119-125, Aug. 2020.
- [12] A. Angdresey, L. Sitanayah, and T. V. N. Kairupan, "A Soil Monitoring and Recommendation System for Ornamental Plants," in *6th International Conference on New Media Studies (CONMEDIA)*, 2021, pp. 40-45.
- [13] A. Angdresey, L. Sitanayah, and T. M. I. Sumajow, "A Real-Time Water Quality and Quantity Monitoring System for Aquarium," in *2021 International Conference on Computer, Control, Informatics and Its Applications (IC3INA)*, 2021, pp. 166-170.
- [14] Z. Wang, Y. Zhou, and G. Li, "Anomaly Detection by Using Streaming K-Means and Batch K-Means," in *5th IEEE International Conference on Big Data Analytics (ICBDA)*, 2020, pp. 11-17.
- [15] H. Fakhi, O. Bouattane, M. Youssfi, and O. Hassan, "New Optimized GPU Version of the K-Means Algorithm for Large-Sized Image Segmentation," in *2017 Intelligent Systems and Computer Vision (ISCV)*, 2017, pp. 1-6.
- [16] S. Daoudi, C. M. A. Zouaoui, M. C. El-Mezouar, and N. Taleb, "A Comparative Study of Parallel CPU/GPU Implementations of the K-

- Means Algorithm," in *2019 International Conference on Advanced Electrical Engineering (ICAEE)*, 2019, pp. 1-5.
- [17] K. Venkatachalam, P. Purandhar Reddy, M. Amudhan, A. Raguraman, and E. Mohan, "An Implementation of K-Means Clustering for Efficient Image Segmentation," in *10th IEEE International Conference on Communication Systems and Network Technologies (CSNT)*, 2021, pp. 224-229.
- [18] S. Chen, A. Shih, and C. Hsiao, "Road Conditions Detection Using Arduino Based Sensing Module and Smartphone," in *IEEE International Conference on Consumer Electronics - Taiwan*, 2015, pp. 254-255.
- [19] K. Chen, M. Lu, X. Fan, M. Wei, and J. Wu, "Road Condition Monitoring Using On-Board Three-Axis Accelerometer and GPS Sensor," in *6th International ICST Conference on Communications and Networking in China*, 2011, pp. 1032-1037.
- [20] A. Mednis, G. Strazdins, R. Zviedris, G. Kanonirs, and L. Selavo, "Real Time Pothole Detection Using Android Smartphones with Accelerometers," in *International Conference on Distributed Computing in Sensor Systems and Workshops (DCOSS)*, 2011, pp. 1-6.
- [21] Kementerian Pekerjaan Umum dan Perumahan Rakyat Direktorat Jenderal Bina Marga. *Manual Desain Perkerasan Jalan (Revisi 2017) Nomor 02/M/BM/2017*.