FINDMYPET: AN INTELLIGENT SYSTEM FOR INDOOR PET TRACKING AND ANALYSIS USING ARTIFICIAL INTELLIGENCE AND BIG DATA

Qinqin Guo[1] and Yu Sun[2]

[1]Portola High School, 1001 Cadence, Irvine, CA 92618
[2]California State Polytechnic University, Pomona,
CA, 91768, Irvine, CA 92620

ABSTRACT

Pet tracking has been an important service in the pet supply industry, as it is constantly needed by countless pet owners [1]. As of 2021, about 90 million families in the U.S. alone have a pet, that is about 70% of all American households. However, for most owners of smaller pets such as cats, hamsters, and more, not being able to find the pet within the house has been a problem bothering them. This paper proposes a tool to use Raspberry Pi for gathering signal strength data of the blue tooth devices and using Artificial Intelligence to interpret the gathered data in order to get the precise location of the indoor moving object [2]. The system is applied to arrive with the location of pets within the house to an accurate level where the room that the pet is located in is correctly predicted. A qualitative evaluation of the approach has been conducted. The results show that the intelligent system is effective at correctly locating indoor pets that are constantly moving.

KEYWORDS

Raspberry Pi, Firebase, machine learning, Artificial Intelligence (AI).

1. INTRODUCTION

It is undoubtedly an annoying and worrying experience for pet owners to mistakenly think that their pet is lost. My household has 4 cats, and a lot of times, some of them hide away under the bed or behind the couch, and I cannot find them for the whole day, so I just keep worrying if they can't get enough food or water [3]. A similar problem occurred when I was actually much younger. At the time, my pet turtle climbed out of the aquarium and was missing for months. By the time I found it under the cabinet, it was already dead. Since then, I always wanted to figure out a way to know where my pets are located all the time. Before I have started this project, I have surveyed pet owners through an online form with random sampling and random assignment, and it turned out that it is a universal concern for pet owners with small animals such as cats, turtles, and reptiles to be unable to identify where the pet is at while know it is at home. My design can help with this common problem among pet holders. With the Find My Pet system, users can quickly identify the specific location of the intended animal as long as the animal is wearing a collar with or attached to a signal emitting device such as a Bluetooth beacon [6]. Therefore, many problems can be solved. For example, cat owners do not need to worry that they cannot find the newly arrived cat who is shy to meet people and always hides away. When pets can be found in a short time, other problems such as dehydration, starvation, and even death can also be prevented.

David C. Wyld et al. (Eds): ARIA, SIPR, SOFEA, CSEN, DSML, NLP, EDTECH, NCWC - 2022
pp. 37-48, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.121504

There are existing domestic animal tracking techniques and systems, such as GPS collars, which allow the users to know the approximate location of the animal [4][5]. However, such designs are only suitable for outdoor tracking where the designated animal may move in great distances, and it rarely works for small-range accurate tracking. For example, the Air Tag collar, commonly used for cats, is a very popular product, but it was not originally designed for pet tracking; Airtags can usually be accurate up to which street and even which household the intended object is, but it is far from enough for the subject in discussion [7]. In addition, Airtag devices often make substantial sounds when being manipulated on the phone to find the lost animal, and this results in further inaccurate results because noises can often scare animals and prompt them to move around even more. Other techniques, such as inserting microchips, have always been used as an identification tool for animals; by inserting an electronic chip with pet and owner's information, the lost animals with microchips can be easily returned to home when found and scanned with a corresponding device. Nevertheless, it is not useful in an active search for the animal because the method was designed mainly to keep the information about the animal and its owner well-organized. There are also animal tracking devices that can be used for precise small range tracking, which are usually Bluetooth devices; however, because the algorithm used cannot be too sophisticated, such devices only show the user the approximate distance between the Bluetooth collar and device that is used to track the Bluetooth collar instead showing the direct location, it can not give enough useful information to locate the pet.

Our goal is to generate an intelligent system that can accurately estimate the location of indoor moving animals. There are some good features of the system. First, the signal emitter that the animal will wear is a Bluetooth device. For the hardware designing and experimenting process, we used Bluetooth iBeacons, and they are proven to be effective for small distance tracking. The iBeacons have relatively low cost and easy accessibility compared to many other similar Bluetooth products, so it is a good tool to be used for this project. Second, Our hardware part includes three Raspberry Pi. Raspberry Pi is a microcomputer that can perform many complex functions including coding, search engineering, and many more. These Raspberry Pi devices are very compatible tools for the project because they are microcomputers that can receive, process, and send signals speedily. Third, we trained artificial intelligence(AI) to interpret the received data and make reasonable and accurate results out of the given data and stored information regarding the data range of specific rooms. This AI is well-adapted for the purpose mentioned above. Last, we use the received number of signals indicator(RNSI) to track locations instead of the commonly used received signal strength indicator(RSSI) because research has found that RNSI shows more significant differences in value when the distance of the moving object from the Raspberry Pi changes more compared to RSSI [8][9]. Also, RNSI is more stable in an environment with high signal interference, where many other irrelevant Bluetooth devices may be present. Therefore, we believe that our model is strong enough to be implemented for the purpose of indoor moving pets tracking.

In the two application scenarios for our tracking system, we demonstrated how the above combination of techniques gives off a useful and accurate prediction model. First, use three BLE beacons and one Raspberry Pi to test the eligibility of the beacons-if they are sending out signals in a consistent way and could that signal be received by our receivers, or the Raspberry Pi. Second, we analyze if the system could produce a precise prediction of the pet's location by using a real pet wearing a beacon for testing. After we obtained RNSI value ranges for each of the rooms in the testing house (single floor), we put the beacon on the experimenting pet(a cat) and placed the cat in a designated room. The results show that our prediction model is accurate. In the 10 trials that we performed, ten out of ten, or 100% of the time the prediction was accurate.

The rest of the paper is organized as follows: Section 2 gives the details on the challenges thatwe met during the experiment and designing the sample; Section 3 focuses on the details of our solutions corresponding to the challenges that we mentioned in Section 2; Section 4 presents the relevant details about the experiment we did, following by presenting the related work in Section 5. Finally, Section 6 gives the concluding remarks, as well as points out the future work of this project.

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. Using Raspberry Pi to properly carry out the process

During our hardware development stage, we encountered challenges while trying to explore how to use Raspberry Pi as a signal receiver and data storage and transferer. We had problems such as the Raspberry Pi was not updated enough for us to carry out certain processes. For example, while looking for tools to detect beacon signals in the library, we were restrained from using the library that incorporated such functions because the version of Raspberry Pi at the time was not capable of running such tools. To use beacon signal detecting tools, we had to manually update the system by running scripts in the Terminal, which took a long time but turned out to be working properly. Another problem was that the Raspberry Pi could not receive the signals sent from the beacons. When we run the script for finding the RNSI value by detecting the signals, the results sometimes turn out to show an error has occurred. This happened many times during the development of the project, especially in the experimenting stage, and we found out that this was caused by when the script started running, the signal from the beacons have not yet been sent, and the will cause the script to interpret the situation as an error and no longer receive signal. To solve this challenge, we added a 60 seconds sleep time at the beginning of the script, so as long as there are signals detected in the first 60 seconds when the scripts start running, the script will continue working and detect the signals.

2.2. Receive accurate RNSI using Raspberry Pi

The signal emitting tools in the project are Bluetooth low energy (BLE) beacons, a class of BLE devices. The BLE beacons broadcast their identifier to neighboring electronic devices through radio waves. However, beacons often do not have a very stable signal emitting pattern; rather, the RSSI may vary even when the beacon is not moved. Using RNSI instead of RSSI has helped, but not much at first because the value still does not make much sense sometimes. Fortunately, we found out that some frequencies of radio waves are more stable compared to others, so we manually adjusted the signal emitting frequency of the beacons through their mobile application. For example, when the frequency of Daisybeaconsmall (the smallest beacon used in the experiment) is adjusted to be twice as high as before, its results turned out to be more accurate and stable: when put done at a certain place and run the programming to receive its signals for 10 times, the RNSI value tends to be the same or 1 or 2 differences from the median value.

2.3. Making an accurate prediction model using the collected data

It is very sophisticated to use machine learning to train an AI to interpret the collected RNSI and end up with the right prediction because the process involves positioning analysis, trigonometry calculations, and calibration. In addition, because there is the interference of signals, it is even harder to interpret the data. To optimize the results from the collected data, we collected a range of data in each of the rooms used for experimenting, so each room (specific area) corresponds with a range of data that consists of three RNSI values, one from each Raspberry Pi. Because the Raspberry Pis are placed in a static position, their distances to each of the rooms will remain constant, so theoretically the RNSI values from a BLE beacon received by the Raspberry Pis will be the same or with small errors as long as the beacon is placed in the same room. This way, we could predict which room is the intended pet (wearing a BLE beacon) based on the range of RNSI values for each of the rooms.

3. Solution

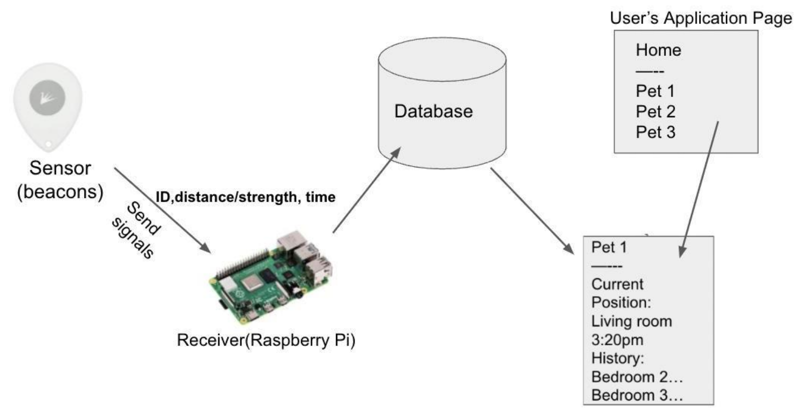

Figure 1. Overview of the solution

FindMyPet is a pet tracking system that is mainly composed of these steps: sending a signal, receiving the signal, processing the signal into useful data, storing data, and giving results based on the perceived data. The sensors are responsible for sending out the signal, they are low-energy Bluetooth (BLE) beacons. These beacons are usually small in size and wearable for animals in a variety of sizes. After the beacons send out radio signals, Raspberry Pis will receive the signals accordingly [14]. The programming language that we use in Raspberry Pi is python, and we programmed the three Raspberry Pis to receive signals every interval of one minute. After receiving signals, Raspberry Pis are programmed to send the RNSI value to Google Firebase where our data is stored and replaced with new data every minute. Then, our Replit program will retrieve the data from Firebase and calibrate it to make a prediction of the location of the pet given that certain values of RNSI from each Raspberry Pi correspond to a specific room of the household. Lastly, the same results that are shown on our Replit website will be shown on the user's application that we developed for this tracking system. When the user clicks "Where is My Pet" on the application side, the whole system of programming will run and give the most recent update to the pet's location. The next section will discuss the details of each of the components mentioned in the project overview.

1. Code on Raspberry Pi-Thonny

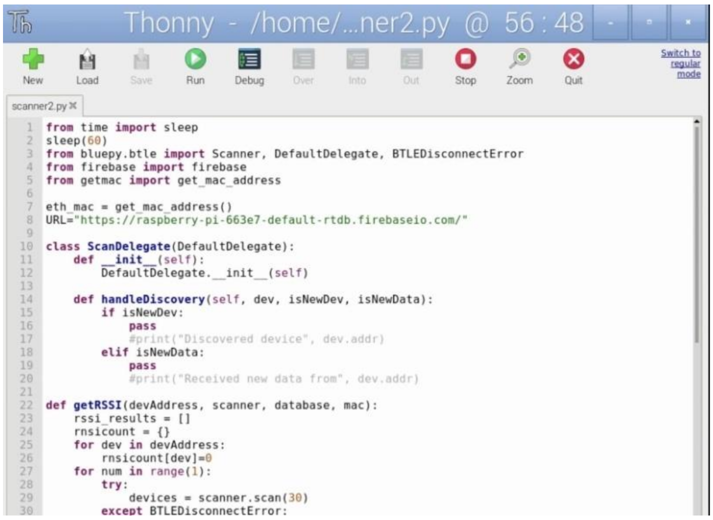

Figure 2. Screenshot of Thonny

On Raspberry Pis, we use python in Thonny to set up the program for receiving signals from BLE beacons and sending RNSI data to Firebase [10]. We added 60 seconds of sleep time before each round of getting data so we would not miss the signals of the beacons when they are just turned on. We imported packages from different python libraries to help us. For example, we import "Firebase" from the "Firebase" library and we import multiple tools from "bluepy.btle".

2. Data on Firebase

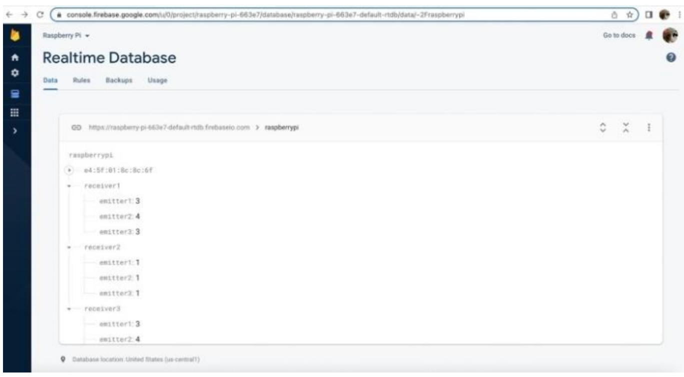

Figure 3. Screenshot of Database

Through Firebase, we receive the RNSI data from the Raspberry Pi. We store and manage the data on this database. The database is named "raspberrypi", and sections under it include receiver 1, receiver 2, and receiver 3, and each has three components of emitter 1, emitter 2, and emitter 3 under them. Raspberry Pi is the receiver and BLE beacons are the emitters. The URL of this database is incorporated into the program in Replit, where the RNSI data will be transferred when the program starts calibrating and making predictions.

3. Code on Replit

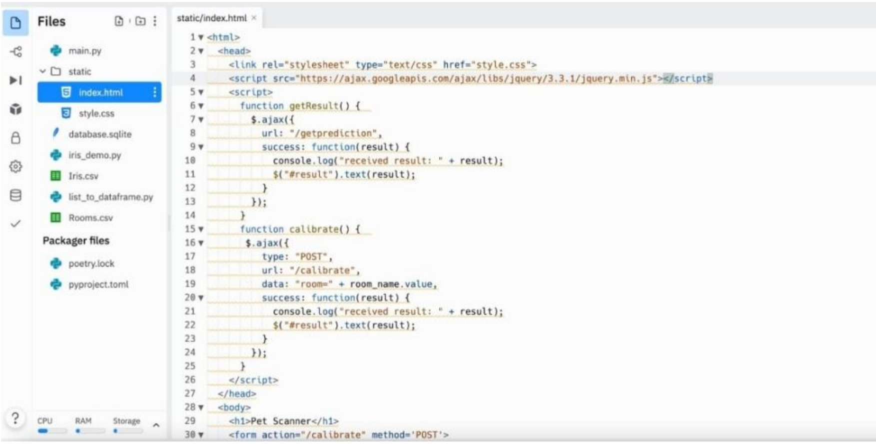

Figure 4. Code on Replit 1

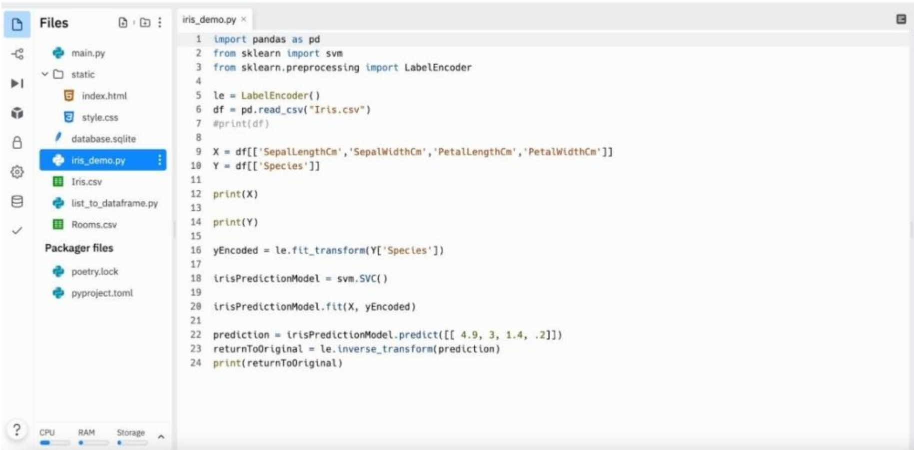

Figure 5. Code on Replit 2

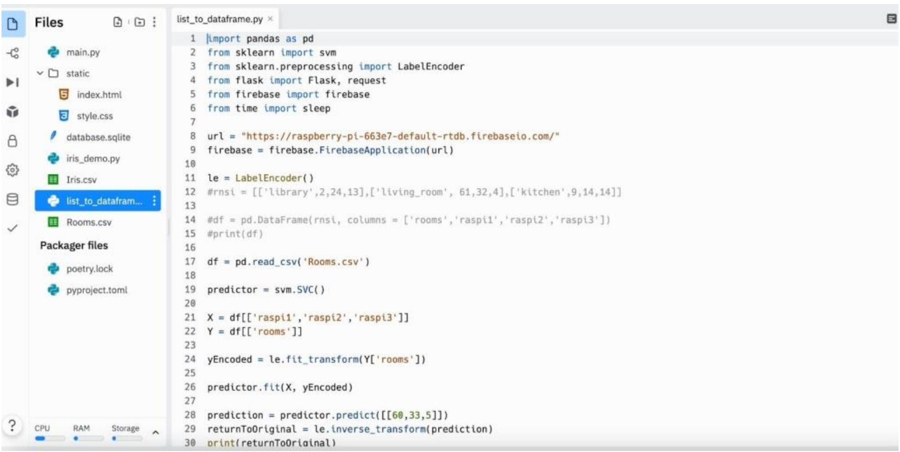

Figure 6. Code on Replit 3

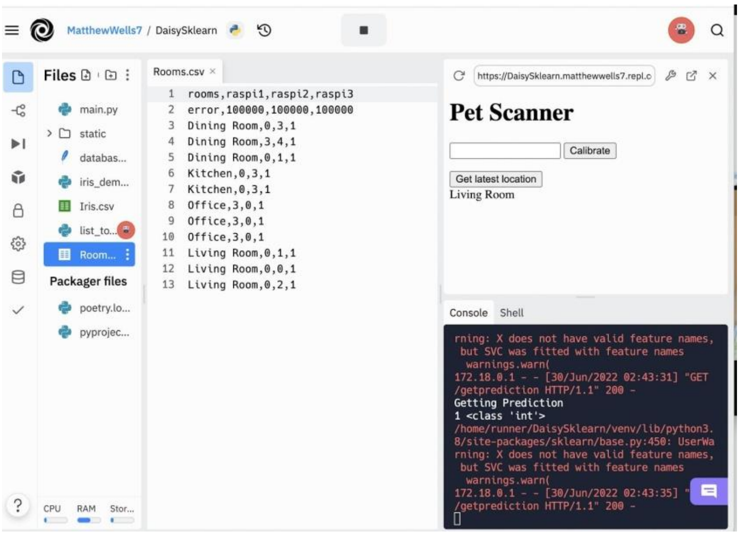

Figure 7. Code on Replit 4

As Figure 4 shows, in “index.html”, we created the web page for scanning. There are basic functions of calibrating, getting predictions, and receiving results. The name of the web is “Pet Scanner”. First, we scan a range of RNSI values for the rooms, which is an input to the calibration. The “getResult” button allows us to get the latest location of the signal emitter, which is the pet wearing the BLE beacon. In Figure 5, here is the code of the package we installed. We use “svm” from the “sklean” library and “LableEncoder” from “sklearn.preprocessing”. We use the Iris.csv package for testing the program. This package allows a function to take in different variables and make a prediction based on the variable ranges. Figure 6 shows “list_to_dataframe.py”. We use the data frame to receive and organize the RNSI data into tables with a range of values corresponding to each of the rooms in the testing area. Figure 7 shows the HTML webpage of the control page with the buttons.

4. Code On Kivy


petfinder.py - /Users/daisyguo/Desktop/coding minds/petfinder.py (3.10.4)

```
from kivy.uix.stacklayout import StackLayout
from kivy.app import App

class ControlPage(StackLayout):
    def calibration():
        pass
    def locate_pet():
        pass

class petfinderApp(App):
    def build(self):
        return ControlPage()

if __name__ == '__main__':
    petfinderApp().run()
```

Figure 8. Code on Kivy 1

petfinder.kv - C:/Users/wells/petfinder.kv (3.10.5)

File Edit Format Run Options Window Help

```
<ControlPage>:
    Label:
        text: Welcome to PetFinder
        text_size: self.width-30, self.height-30
    TextInput:
        id: room_name
    Button:
        id: calibrate
        on_press: root.calibration()
    Button:
        id: locate
        on_press: root.locate_pet()
    Label:
        id: output
```

Figure 9. Code on Kivy 2

We use Kivy to create a desktop user interface and mobile application for the PetScanner users. The application consists of basic functions and buttons. On the control page, we have a calibration button and a pet button. The output will be shown on the same page.

4. EXPERIMENT

4.1. Experiment 1

Design Experiment: In the first experiment, we want to prove the eligibility of the BLE beacons that are used in the project. To do so, we used three beacons and one Raspberry Pi (#1, or receiver 1). The goal of the experiment is to show that all three beacons are working properly (sending signals). There are ten trials, which is enough sampling. In each trial, we placedthe three beacons near each other in one room/one area of the testing area, and moved them fromplace to place each trial.

Trial	Emitter 1	Emitter 2	Emitter 3
1	2	2	2
2	6	8	7
3	9	8	8
4	10	13	11
5	4	4	4
6	5	8	5
7	10	17	10
8	4	7	5
9	6	6	6
10	5	7	6

Figure 10. Experiment 1 trial and results

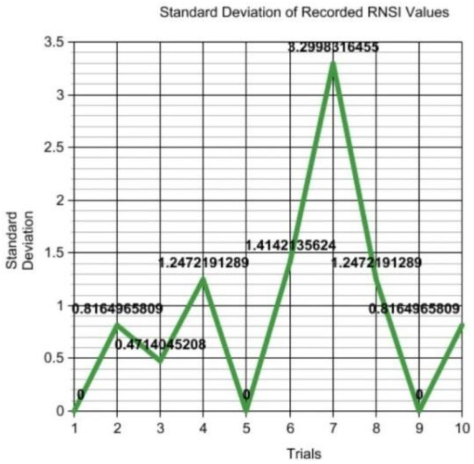

Figure 11. Result of experiment 1

Summary: The results of the experiment shows that the three beacons are functioning properly. The three beacons are each assigned a column, and they are named in this case, Emitter 1, Emitter 2, and Emitter 3. All three of the beacons are sending out signals that are receivable by the Raspberry Pi. In each trial, the number below each of the emitter columns is the corresponding RNSI value of the beacon in that trial. For trial 1, 5, and 9, all three emitters resulted in the same RNSI value, and in other cases, the RNSI value differences for emitters in the same trial range from 1-7, and this shows that the beacons are consistent in sending signals. There is also a graph showing the standard deviation of the standard deviation(SD) values for each trial, with the SD values ranging between 0 to 3.2998316455. The SD values are relatively small, and this means that the RNSI values are clustered around the mean value, or they are less spread out.

4.2. Experiment 2

Design Experiment: In the second experiment, we used the same beacon/same pet to test at the same location for 10 times to test the accuracy of the prediction model. In this experiment, a real pet, in this case, a cat, wore a collar with a BLE beacon, and was released to move freely around the testing area (the house that the cat usually lives in). After 10 minutes, we started to run the program to get predictions regarding the cat's location. We consistently run the program for 10 times in a row in a short time, to see if the results are consistent (same), and in the end, if they are accurate.

```
Dining Room,0,3,1
Dining Room,3,4,1
Dining Room,0,1,1
```

```
Dining Room,1,3,1
Dining Room,1,3,1
Dining Room,2,2,1
```

(Example of calibration for a room)

Figure 12. Experiment 2 examples

Trial	Result
1	Dining Room
2	Dining Room
3	Dining Room
4	Dining Room
5	Dining Room
6	Dining Room
7	Dining Room
8	Dining Room
9	Dining Room
10	Dining Room

Figure 13. Result of experiment 2

The order is arranged in: room name, rasp 1, rasp 2, rasp 3, where each Raspberry Pi' s RNSI value is shown. The above figure shows the RNSI values that are calibrated to correspond with the room "Dining Room". It is stored and updated in Firebase and shown in Replit. As the figure shows, there is a set of values that correspond with this room (note: RNSI values are only shown in integers). We performed over 20 calibrations for each of the rooms we used in the experiment, and the example shows that of the Dining Room. After getting the RNSI value for each room, we placed the pet with the beacon collar in the Dining Room and ran the program to see if the prediction would be accurate. The results turned out to be 100% accurate, with ten out of ten predictions to be "Dining Room", which is where the pet with the beacon is actually located.

The experiment results for both experiments show that our system is functioning according to our expectations. In the first experiment, we have shown that the Beacons are consistently working and sending signals, and when three beacons are placed together, they will have similar RNSI values. This shows that the challenge of inconsistent RNSI value is partially solved, that at least we know the three beacons are responding with signals and the signals are properly catched by the Raspberry Pi. In the second experiment, the results show that the calibration is functioning well, and the artificial intelligence used for prediction is also very accurate. This indicates that our algorithm is useful at indoor location prediction and it was a good choice to use RNSIinstead of RSSI because RNSI gives accurate results according to our experiment. The challenges have been mostly solved because Raspberry Pi is carrying out the process correctly, the RNSI values are producing meaningful results, and our prediction model using the RNSI datais giving accurate predictions.

5. Related Work

In the research paper "tracking a moving user in indoor environments using Bluetooth low energy beacons", the researchers discussed their approach of using RNSI-based location tracking system instead of the commonly used RSSI-basd tracking approach and proved that RNSI has a more accurate reflection of the object's location in an indoor location compared to RSSI [11]. The researchers want to use the results to further the study of tracking human movements, especially in the setting of healthcare locations where high levels of signal interference are present and the environment is very dynamic. The big difference between the research of the aforementioned paper and our research is that our research focuses on the use of RNSI-based location tracking systems for animal tracking in common households, where the area is usually significantly smaller than in healthcare places(where the researchers of the other paper did their experiments) and has less signal interference factors.

In the research paper “Protection of the Child/Elderly/Disabled/Pet by Smart and Intelligent GSM and GPS based Automatic Tracking and Alert System", researchers have developed a tracking system where they use the existing GSM network and GPS satellites [12]. This approach focuses on such a method because it is possible to implement their system on a large scale at a relatively low cost compared to many other tracking approaches. Both this research and our research is aimed at developing tracking systems for moving individuals, whether it’s for animals or humans, or even vehicles. The big difference between our work is that the aforementioned research emphasized large range tracking while mine focused on small range tracking. For example, their paper discusses the purpose of their research, which includes preventing the kidnapping of children, loss of soldiers, cognitive difficulties for elders and mentally disabled people, and more.

In “Detepet Mobile Application for Pet Tracking”, researchers from Bina Nusantara University discussed their new application developed for pet tracking and extended pet care services [13]. Their application allows GPS tracking of pets with GPS collars, forums to post lost pet information, an online pet supply store, and information for pet-related events. Their tracking system is very mature and has the extended function of keeping track of the footprints of animals; by calculating the footprints using the size of the animal, the owner can know more about the health status of the pets by knowing how much the pets have exercised. This system is intended for large range animal tracking to prevent the animal loss, while it could work at a small range, though it was not designed for small range indoor tracking. My system is designed and calibrated for accurate small-scale indoor environment pet tracking.

6. CONCLUSIONS

In our work, we designed a system named “FindMyPet” to make accurate indoor location predictions for moving objects (primarily pets) using RNSI values interpreted by artificial intelligence. The main components of the system include BLE beacons, Raspberry Pis, Firebase, Replit, and Kivy. The beacons are responsible for sending out signals and Raspberry Pis are responsible for receiving the signals and making them into RNSI values. The RNSI value is calculated through artificial intelligence in Raspberry Pi and is sent to Firebase to be stored and managed [15]. Then, the RNSI data will be used by the python program on Replit for calibration and calculating the location. In the end, users of the system can access the system through a mobile application or desktop user interface. We performed two experiments using the established system: 1. using one Raspberry Pi and three beacons to test the eligibility of the beacons. 2. Having a moving pet wearing the beacon to test the accuracy of the current prediction model. Both experiment results indicate that the system is effective and have solved the major challenges. The beacons are working properly and the prediction system is mostly accurate at indoor tracking.

Currently, there are still a few limitations regarding this system. First, in households with multiple electronic devices, the RNSI values may be interfered with to produce inaccurate results. Common objects such as microwaves, which can release electromagnetic radiation, can cause significant flaws in receiving signals from beacons. Second, the current system is moderately complicated to set up. If it is sold as a product in the future on the pet service market, it might cause some difficulties for the users because in order to use the system correctly, the user needs to first record the RNSI value range for each of the rooms properly and store in the administration section of the APP, and technical difficulties in setup may occur. Lastly, the current system only works with single-floor households because it uses three Raspberry Pi and trigonometry analysis. For multi-floor households, the application of this system is largely limited.

There are possible future works that could solve the current difficulties. To reduce as much signal interference as possible, we could test a wider range of signal frequencies. Signal frequencies that are significantly different will make it easier for the receivers to filter out the untargeted signals. If the system is too hard to set up, we could offer straightforward instructions and customer service to help. Very importantly, by using four Raspberry Pi instead of three, we could achieve pet tracking at multi-floor households, and that is something we plan to start in the near future.

REFERENCES

[1] Lin, Cindy Xide, et al. "Pet: a statistical model for popular events tracking in social communities." Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining. 2010.
[2] McCarthy, John. "What is artificial intelligence." URL: http://www-formal. stanford. edu/jmc/whatisai. html (2004).
[3] Blouin, David D. "Understanding relations between people and their pets." Sociology Compass 6.11(2012): 856-869.
[4] Johnson, Chris J., Douglas C. Heard, and Katherine L. Parker. "Expectations and realities of GPS animallocation collars: results of three years in the field." Wildlife Biology 8.2 (2002): 153-159.
[5] Cagnacci, Francesca, and Ferdinando Urbano. "Managing wildlife: a spatial information system for GPScollars data." Environmental Modelling & Software 23.7 (2008): 957-959.
[6] Chawathe, Sudarshan S. "Beacon placement for indoor localization using bluetooth." 2008 11th International IEEE Conference on Intelligent Transportation Systems. IEEE, 2008.
[7] Haskell-Dowland, Paul. "Remember, Apple AirTags and 'Find My' app only work because of a vast, largely covert tracking network." The Conversation (2021).
[8] Parker, Ryan, and Shahrokh Valaee. "Vehicular node localization using received-signal-strength indicator."IEEE Transactions on Vehicular Technology 56.6 (2007): 3371-3380.
[9] Liu, Chong, Kui Wu, and Tian He. "Sensor localization with ring overlapping based on comparison of received signal strength indicator." 2004 IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE Cat. No. 04EX975). IEEE, 2004.
[10] Annamaa, Aivar. "Introducing Thonny, a Python IDE for learning programming." Proceedings of the 15th Koli Calling Conference on Computing Education Research. 2015.
[11] Surian, Didi, et al. "Tracking a moving user in indoor environments using Bluetooth low energy beacons." Journal of Biomedical Informatics 98 (2019): 103288.
[12] Punetha, Deepak, and Vartika Mehta. "Protection of the child/elderly/disabled/pet by smart and intelligent GSM and GPS based automatic tracking and alert system." 2014 International conference on advances in computing, communications and informatics (ICACCI). IEEE, 2014.
[13] Aqraldo, Brian Wijaya, et al. "Detepet mobile application for pet tracking." 2021 International Conference on Emerging Smart Computing and Informatics (ESCI). IEEE, 2021.
[14] Parsons, J. D., and A. M. D. Turkmani. "Characterisation of mobile radio signals: model description." IEE Proceedings I (Communications, Speech and Vision) 138.6 (1991): 549-556.
[15] Moroney, Laurence. "The firebase realtime database." The Definitive Guide to Firebase. Apress, Berkeley, CA, 2017. 51-71.