Smart Campus IoT Guidance System for Visitors Based on Bayesian Filters

Alvaro Aspilcueta Narvaez, Dennis Núñez Fernández, Segundo Gamarra Quispe, and Domingo Lazo Ochoa

Abstract In this work, we proposed an indoor location system that makes use of a Raspberry Pi embedded computer and WiFi signals to guide a person inside a region of the faculty of Electrical and Electronic at Universidad Nacional de Ingeniería, Peru. The main advantage with similar indoor location systems like beacons or RFID technology is that the presented system does not require additional hardware since it makes use of the pre-installed WiFi routers. The experimental tests show promising results, achieving a location accuracy of 92.31%.

Keywords Indoor location · WiFi · Bayes filter · Embedded computer

1 Introduction

The existing navigation systems like Global Positioning System (GPS) offer a precise location in outside environments but an inaccurate location inside common buildings [1]. This problem of localization has attracted the interest of researchers and developers due to the high demand for such systems for indoor navigation, immersive experiences, asset tracking, augmented reality, and more.

During the last decade, several approaches for indoor location systems [2] such as radio frequency identification (RFID), wireless local area networks (WLAN), and Bluetooth among others have been proposed [3]. In [4, 5], an indoor positioning

A. Aspilcueta Narvaez · D. Núñez Fernández (✉) · S. Gamarra Quispe · D. Lazo Ochoa
Universidad Nacional de Ingeniería, Lima 15333, Peru
e-mail: dnunezf@uni.pe

A. Aspilcueta Narvaez
e-mail: alvaro.aspilcueta.n@uni.pe

S. Gamarra Quispe
e-mail: sgamarraq@uni.edu.pe

D. Lazo Ochoa
e-mail: dlazo@uni.edu.pe

Y. Iano et al. (eds.), *Proceedings of the 5th Brazilian Technology Symposium*, Smart Innovation, Systems and Technologies 202,
https://doi.org/10.1007/978-3-030-57566-3_46

system for smart buildings was used to obtain relatively good results; however, the system makes use of RFID technology, which produces an additional cost for hardware implementation. Regarding indoor location using WiFi signals, [4] makes use of Monte Carlo (MC) filter for a precise WiFi-based indoor localization, but the system is intended for tracking objects with a high precision employing relative high computing processing, which is not suitable for real-time applications since it demands high computing resources and a small division of the regions. In recent work, [6, 7] proposes a novel, incremental approach that reduces the energy consumption of WiFi localization by scanning just a few selected channels. The results are remarkable and provide a way for the implementation of indoor location on embedded systems. In addition, Indoor Google [8] provides guidance; however, its accuracy is not enough (5–15 m).

In this paper, we propose an indoor location system for a region of the faculty of Electrical and Electronic of Universidad Nacional de Ingeniería, Peru. Unlike the indoor location systems presented before, which rely on external devices like RFID hardware or Bluetooth beacons, the proposed system does not demand external equipment or additional costs. Therefore, employing an embedded computer and pre-installed access points, our indoor location system is able to provide a good position accuracy and the guidance to the destination region.

2 Proposed Method

2.1 Overview

Our system performs indoor localization and guidance on an embedded computer using WiFi signal levels. In this way, the system works on small embedded devices with low computational resources. With the aim to achieve response time and power computational constraints, a simple and effective Bayes recursive estimator was used in this work.

The methodology used for this work has two phases: offline and online phase. At the offline phase, the WiFi levels were captured several times in every region, thus obtaining a received signal strength indicator (RSSI) table per region. Later, the RSSI data were divided into training and testing datasets in order to perform the evaluation. Then, the normalized histograms are cleaned using the Gaussian distribution. Finally, the localization system is evaluated offline using average accuracy and confusion matrix. At the online phase, the Bayesian filter was implemented in a Raspberry Pi embedded computer. This filter uses the captured WiFi signal levels and the cleaned RSSI database to predict the most likely region. Figure 1 shows the diagram of the proposed methodology.

2.2 Data Recollection

First, the principal regions of the faculty of Electrical and Electronic of Universidad Nacional de Ingeniería are identified and divided into several areas. Figure 2 shows

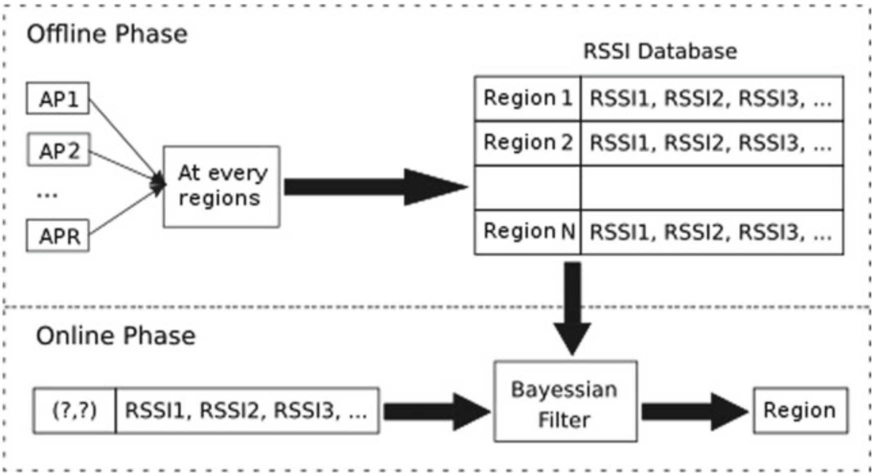


Fig. 1 Scheme of the proposed interior locating system

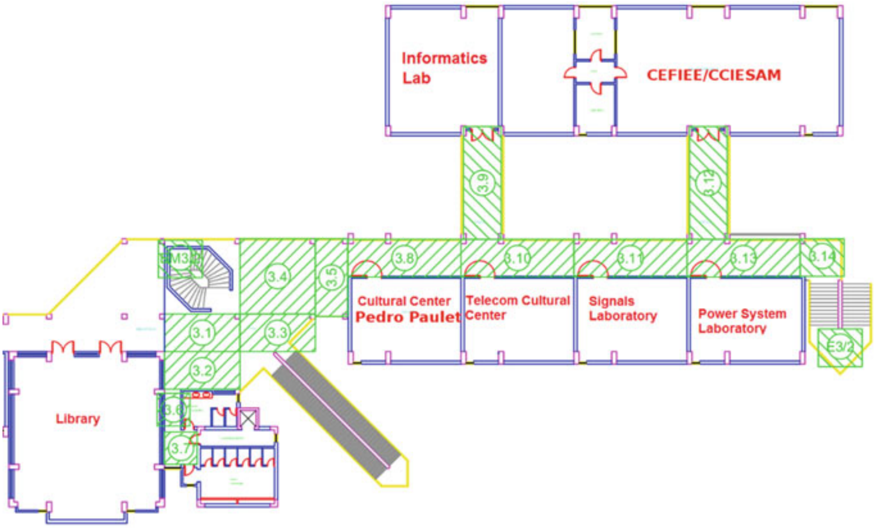


Fig. 2 Sampling points and destination in the third-floor scheme

such a distribution of regions. After division, the main WiFi information like service set identifier (SSID), media access control (MAC) and RSSI is recollected via a custom application in for the Raspberry Pi. Thereby, 150 samples in each region at different times during three non-consecutive days/nights were captured. Finally, filtering processing for non-stable access points (APs) was carried out, so the final dataset was constructed using only the best APs in order to have the best results.

2.3 Bayesian Filter

Several relevant papers about RSSI-based indoor location systems assume Gaussian probability density function (PDF) [9, 10]. It is justified by the relation of the noise of the radio receiver together to the influence of average white Gaussian radio channel noise, which is modeled by a Gaussian PDF. Therefore, original histograms could be cleaned by approaching each histogram to a Gaussian PDF. In Eq. (1), the formulation of a Gaussian PDF is presented. It is described by two parameters: μ (mean) and σ (standard deviation).

$$f(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{1}$$

In order to achieve response time and computational requirements, a straightforward and effective estimator is used, and it is based on the Bayes recursive estimator. This is able to infer the posterior using sensed and prior knowledge (see Eq. (2)). Being *A* the event we want the probability, and B the new evidence that is related to *A*. Therefore, the posterior *P(A|B)* is calculated by the likelihood *P(B|A)* (probability of observing the new evidence) and the prior *P(A)* (probability of our hypothesis without any additional prior information). *P(B)* is the marginal likelihood, which is the total evidence probability.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{2}$$

The algorithm presented below corresponds to the implementation of the Bayes-based estimator. This takes the AP tables and current WiFi measurement as inputs and returns the estimated region and its probability. Thereby, the algorithm recursively calculates the probability of the posterior region (line 11), and then, the probability (line 12) and predicted region (line 13) are calculated. Algorithm 1 was implemented in Python for evaluation of the performance of the Bayes estimator and for inference on the Raspberry Pi platform.

Algorithm 1 Bayes Estimator Algorithm

$N = number\ of\ regions;\ R = number\ of\ routers$
$W_r = AP\ table\ for\ router\ r$
procedure BayesEstimator($w_1, w_2, ..., w_R$)
    \# Start with uniform distribution
    $priorW_{1,2,..,R} = [1/N; 1/N; ...; 1/N]_{Nx1}$
    $probability = (100/N)\%$
    while probability $<$ 95% **do**
        \# Perform Bayes
        for r from 1 to R **do**
            $posteriorW_r = norm(priorW_r \times W_r[:, w_r])$
            $prob_r = max(posteriorW_r)$
            $pred_r = where(posteriorW_r == prob_r)$
        end for
        \# Find the highest probability
        $probability = max(prob_{1,2,..,R})$
        $r\_best = where(prob_{1,2,..,R} == probability)$
        $prediction = pred_{r\_best}$
        \# Update the new prior
        for r from 1 to R **do**
            $priorW_r = posteriorW_{r\_best}$
        end for
    end while
    Return $prediction, probability$
end procedure

2.4 Guidance System

The flowchart of the operation of our guidance system is shown in Fig. 3. The start screen allows the manual entry of the destination point through a drop-down menu, based on the plane of sampling points shown in Fig. 2. Once selected, the user pushes the central button in order to start the guidance system. The destination region is displayed on the screen.

Then, the system calculates the current node where the user is located, which is the returned result of the indoor location algorithm based on the Bayes filter, see Fig. 3. Based on the current node and the destination node initially entered, the graph of the nodes is expressed in an adjacency matrix of the distances and angles between the nodes (based on the plane of the faculty facilities). Then, the algorithm of Dijkstra calculates the shortest route and returns the next node to go, which is based on the minimum route.

From the provided map of the facilities, the orientation with respect to the magnetic north is also shown, and the use of a magnetometer was added to indicate the orientation in degrees from the magnetic north; so, it works like a compass, no matter the direction of the user it will always orient to the north direction. In order to have the same direction, it was previously calibrated. Some considerations were taken such as a small variation because the screen generates a magnetic field, see Fig. 4.

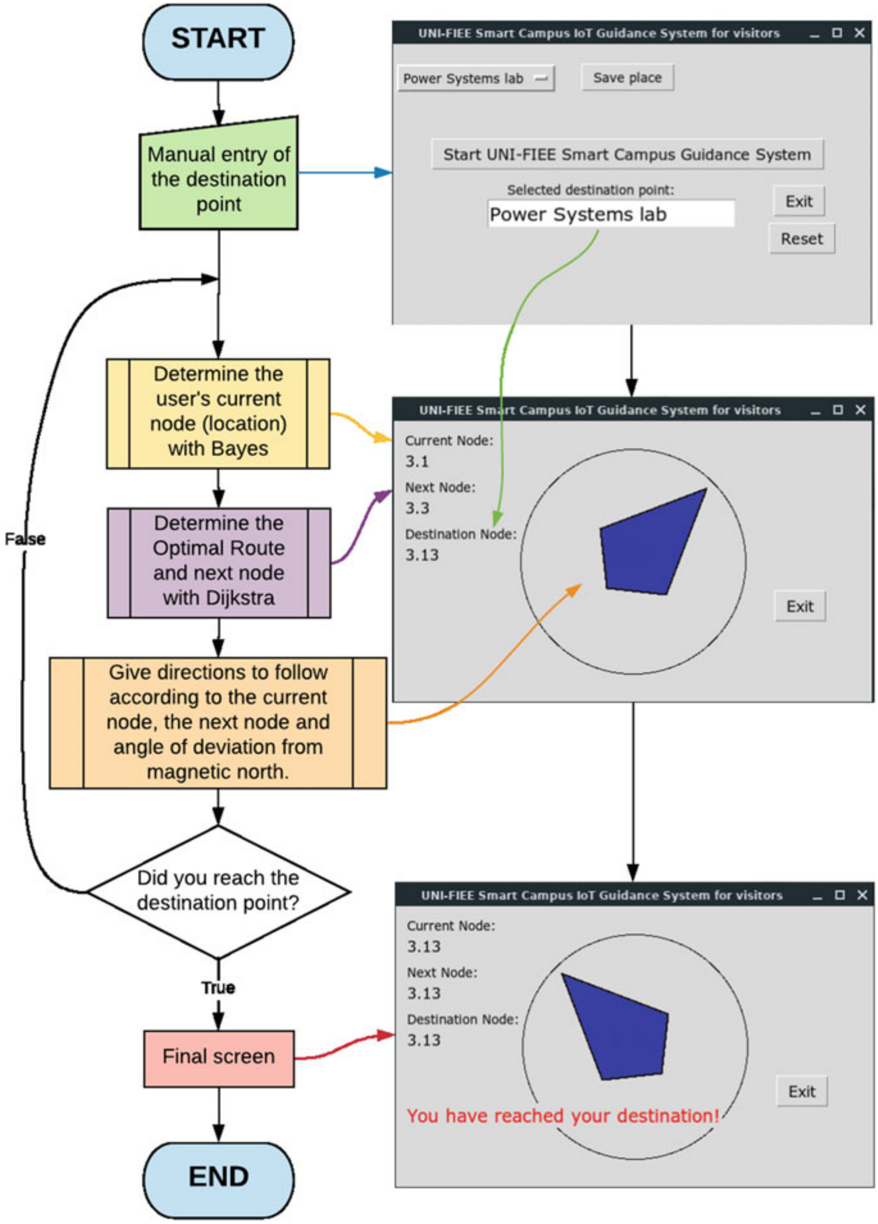
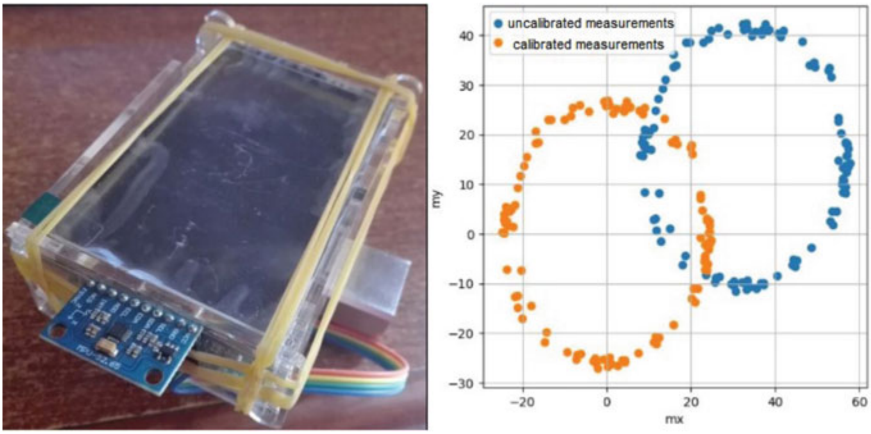


Fig. 3 Flowchart of the smart campus guidance system program



(a) Location of the magnetometer (b) Calibration of magnetometer

Fig. 4 Hardware and calibration related to the magnetometer

Using the angle with respect to the magnetic north, and the angle between the current node and the next node, which was determined by the Dijkstra algorithm, the direction of the guiding compass is independent of the direction of the user to the next node, which follows the minimum route to reach its destination.

When the user arrives at their final destination, the system finishes working and the next short message will appear on the screen of the Raspberry Pi board: "*You have arrived at your destination!!!*".

3 Experimental Results

In order to obtain the performance and other metrics for the indoor localization system, we evaluated it on the testing dataset. The overall accuracy and the confusion matrix (see Fig. 5) define the results. This matrix for the proposed system shows high accuracy for the diagonal values and low accuracy for the others, and it means that the system has a high accuracy per class/region. In addition, the accuracy of the indoor location system is about 92.31%.

For the evaluation of the system, we use a Raspberry Pi model 3B+ , an LCD 3.5″ screen, an MPU9250 sensor, and an external battery. It was conducted in different regions and several positions. As expected, the system correctly predicted the locations in all regions. Figure 6 shows some results of the test. As we can see, our system correctly predicts the location.

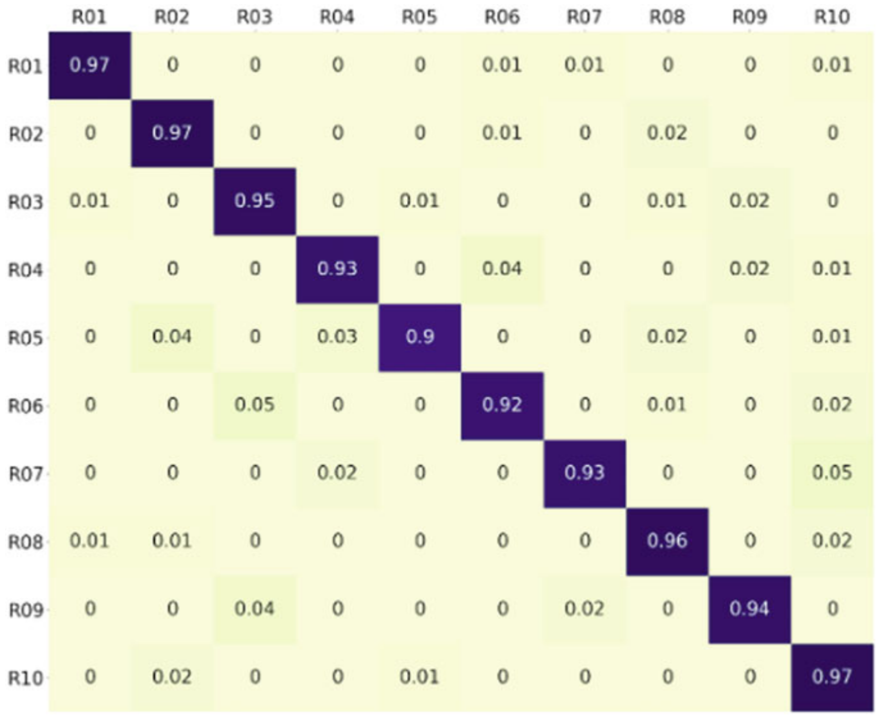


Fig. 5 Confusion matrix for the proposed system

4 Conclusions

This paper introduced the implementation of a Smart Campus WiFi-based indoor location system to guide visitors in a region of the faculty of Electrical and Electronic of Universidad Nacional de Ingeniería, Peru. With the aim to accomplish a high positioning accuracy and a low computational power consumption, the proposed system makes use of Bayes recursive estimator. Experiments show promising results, obtaining an accuracy of 92.31%. As further work, the processing could be performed in remote servers in order to have a better response time. Thus, the proposed methodology is not limited to this work, but can also be applied to similar localization tasks such as immersive experiences, augmented reality, and asset tracking, among others.

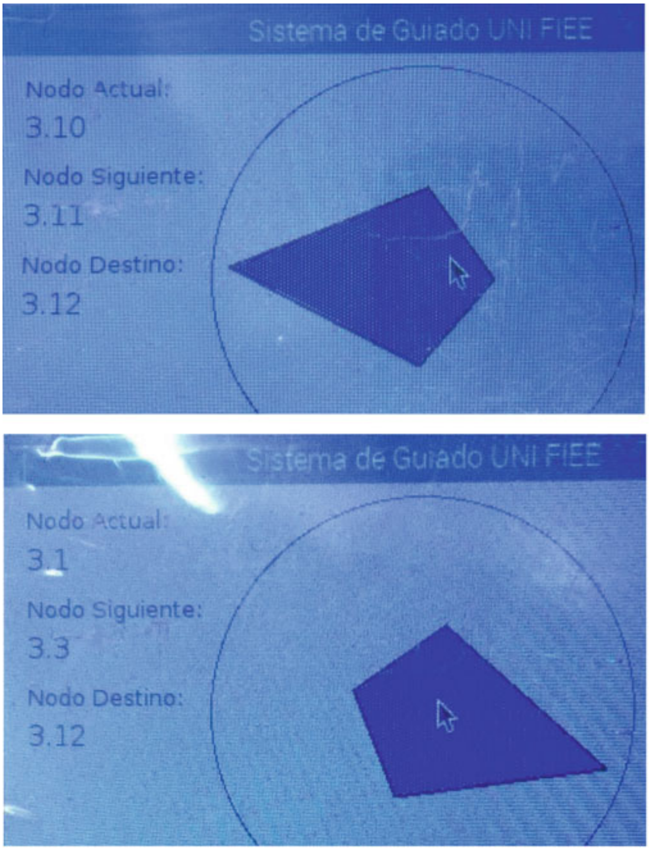


Fig. 6 Online results on the Raspberry Pi. **a** Result in region 3.10. **b** Result at region 3.1

Acknowledgements We thank Ph.D. Marco Zúñiga Zamalloa for their time and inspiring teaching. This work was funded by the Research Institute of the Faculty of Electrical and Electronic Engineering (IIFIEE UNI).

References

1. Motte, H., Wyffels, J., De Strycker, L., Goemaere, J.-P.: Evaluating GPS data in indoor environments. Adv. Electr. Comput. Eng. **11**(3), 25–28 (2011). https://doi.org/10.4316/AECE.2011.03004
2. Sakpere, W., Adeyeye Oshin, M., Mlitwa, N. B.: A state-of-the-art survey of indoor positioning and navigation systems and technologies. S. Afr. Comput. J. **29**(3), 145–197 (2017)
3. Brena, R. F., García-Vázquez, J. P., et al.: Evolution of indoor positioning technologies: a survey. J. Sensors **2017**, 21 (2017)

4. Dhital, A., Closas, P., Fernández-Prades, C.: Bayesian filters for indoor localization using wireless sensor networks. In: 2010 5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC), pp. 1–7. Noordwijk (2010)
5. Moreno-Cano, M. V., Zamora-Izquierdo, M. A., Santa, J., Skarmeta, A. F.: An indoor localization system based on artificial neural networks and particle filters applied to intelligent buildings. Neurocomputing **122**, 116–125 (December 2013)
6. Brouwers, N., Zuniga, M., Langendoen, K.: Incremental Wi-Fi scanning for energy-efficient localization. In: 2014 IEEE International Conference on Pervasive Computing and Communications (PerCom), pp. 156–162. Budapest (2014)
7. Lin, K., Chen, M., Deng, J., Hassan, M. M., Fortino, G.: Enhanced fingerprinting and trajectory prediction for IoT localization in smart buildings. IEEE Transact. Autom. Sci. Eng. **13**(3), 1294–1307
8. Indoor Google Maps. (Online Web, accessed on January 2014). Available: http://maps.google.com/help/maps/indoormaps/
9. Chruszczyk, L.: Statistical Analysis of Indoor RSSI Read-outs for 433 MHz, 868 MHz, 2.4 GHz and 5 GHz ISM Bands. Int. J. Electr. Telecommun. **63**(1), 33–38 (2017)
10. Kaji, K., Kawaguchi, N.: Design and implementation of WiFi indoor localization based on Gaussian mixture model and particle filter. In: 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN), pp. 1–9. Sydney, NSW (2012)