HanDomotic: a Gesture-Based Domotic Control System

Lorenzo Ceccanti
l.ceccanti4@studenti.unipi.it

Cristiano Corsi
c.corsi9@studenti.unipi.it

Marco Imbelli Cai
m.imbellicai@studenti.unipi.it

Niccolò Mulè
n.mule1@studenti.unipi.it

ABSTRACT

This paper presents HanDomotic, a gesture-based control system for home automation which employs Bluetooth Low Energy (BLE) beacons and wearable technology. The system integrates an Android smartphone and a WearOS smartwatch to recognize hand gestures within an indoor environment. BLE beacons are utilized for accurate room localization by analyzing the Received Signal Strength Indicator (RSSI), offering a low-energy alternative to GPS and Wi-Fi-based solutions.

The core functionality involves detecting gestures to control home appliances. A Support Vector Machine (SVM) classifier, trained on tri-axis accelerometer data, identifies gestures such as "Circle" and "Double Clap". The system achieves perfect accuracy for "Circle" and "No Gesture", and 88% accuracy for "Double Clap", with a 12% misclassification rate.

The hardware setup includes BLE beacons, an Android smartphone for configuration and communication purposes, and a WearOS smartwatch for continuous monitoring and gesture recognition. This ensures a user-friendly setup and reliable performance.

HanDomotic advances mobile systems by providing an efficient and accurate solution for gesture-based home automation. Future work will aim to improve classification accuracy for complex gestures and add the necessary logic to control smart domotic devices.

1 INTRODUCTION

This project work consisted in developing a set of applications (an Android smartphone application and a Smartwatch Wear-OS application) with the aim of recognizing hand gestures in an indoor environment. As it's known in indoor positioning literature, localizing an user in an indoor environment could be a challenge if using instruments like GPS, mainly because of the lack of coverage inside a building and the high energy consumption required by a mobile device to capture such signal. The scenario that has been considered during the development process consisted in positioning a number of Bluetooth Low Energy Beacons inside a private home. The signal emitted by the Beacons' antennas was used to determine which room the user could be localized in. In particular, we have considered the Received Signal Strength Indicator (RSSI) to infer the user's position inside a specific room. We have decided to use this approach rather then other techniques like Wi-Fi fingerprinting mainly for three reasons:

(1) BLE consumes much less energy compared to IEEE 802.11 Wi-Fi
(2) Wi-Fi fingerprinting would require a too complex knowledge of Machine Learning techniques [2] which are not necessary in our context (we only need to know if the user is inside a room or not).
(3) BLE uses a multi-frequency channel hopping mechanism in order to avoid possible interferences generated by other wireless devices (home appliances, Internet Access Point generally operate on the 2.4 GHz spectrum)

Once the indoor location of the user has been retrieved, we proceeded with the next step: detecting hand-gesture. The idea is the following: the user enters in a room, makes a gesture and expects that a certain action *is actuated* (e.g.: conditioner turning on, invert the status of a smart bulb, . . .). To do so, we tried to gather some data while performing two kind of wrist gestures (here referred to as *Circle* and *Double Clap*, fig: 1) in order to train a Support Vector Machine classifier as suggested by *L.Porzi et al.* [1]

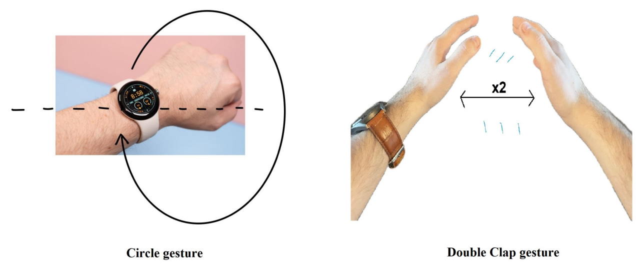

Figure 1: The two kind of gesture considered

2 ARCHITECTURE

2.1 Hardware architecture

In order to develop our system, we adopted the following hardware setup:

(1) Kontakt Bluetooth iBeacons, that will allow our app to detect in which room the user is located;
(2) An Android smartphone, used to host a companion app that scans for beacons, allows the user to register the rooms where the beacons are placed and push the beacon configuration to the smartwatch, which must be previously paired to the smartphone. The app simplifies the configuration process, which would otherwise be uncomfortable to be carried out on a smaller device like a smartwatch;
(3) A smartwatch with WearOS, running our core application, responsible for gesture recognition and localization retrieval.

2.2 Software Workflow

The software component of our system operates through the following steps:

(1) The app running on the smartwatch periodically checks for nearby known beacons, keeping a list of them ordered by RSSI;
(2) The smartwatch regularly samples tri-axis accelerometer data, which is kept in a buffer. This buffer, which has a fixed size, acts as a sliding window over our accelerometer data;
(3) Based on a set frequency parameters, the WearOS app extracts specific features from the accelerometer data and runs a classification using a Support Vector Machine (SVM) model that we have previously trained in a Python environment and exported in a ONNX format;
(4) The app applies a threshold parameter to determine the minimum confidence required by the SVM model to classify the accelerometer features as a specific gesture;
(5) Upon gesture detection, the WearOS app retrieves the closest beacon and its associated room, thus identifying which gesture was made in which location.

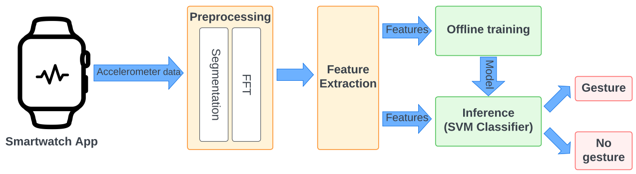

Figure 2: Software workflow schema

3 EXPERIMENTAL RESULTS

3.1 Discussion on classification process

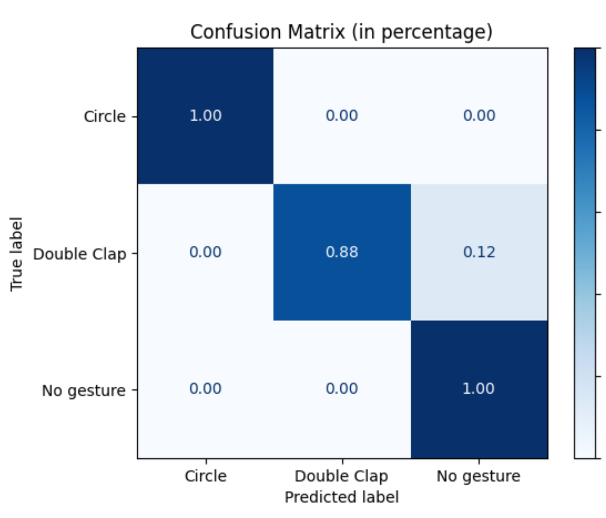

Figure 3: Confusion Matrix

The confusion matrix (3) reveals that the classifier is highly effective in recognizing "Circle" and "No Gesture" states, achieving perfect accuracy in both categories. However, the "Double Clap" gesture shows a lower accuracy (12% misclassification rate), indicating a need for further refinement in distinguishing this specific gesture.

A possible reason for this misclassification could be that, due to the rapidity and sharpness of the "Double Clap" movement, the moving window we used is too large (50 samples, with a frequency of 50 Hz); therefore we obtain a much better performance with the "Circle" movement, which is slower and smoother than the "Double Clap" one.

3.2 Discussion on energy consumption

As energy consumption concerns, we performed a simple energy consumption analysis by using the *AccuBattery* application. This application is able to return the battery consumption and the discharge rate in a rather simple way, without needing to use tools like *Perfetto* or *Battery Historian* that, as far as the experience with our devices is concerned, require having devices with *root permissions* unlocked.

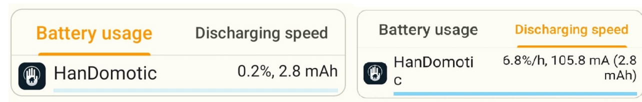

Figure 4: Consumption of smartphone app in foreground

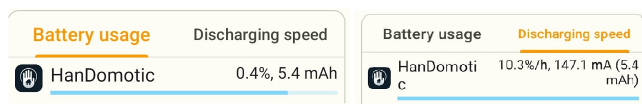

Figure 5: Consumption of smartphone app while scanning

This first set of images (fig: 4, 5) shows the results obtained when leaving the application in foreground and when the app performs the BLE iBeacon scanning process, respectively. We can see that the latter doesn't affect too much the energy consumption compared to simply having the app opened.

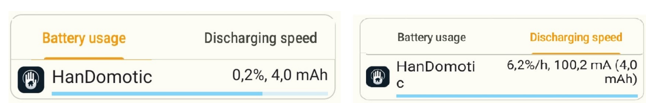

Figure 6: Consumption of smartwatch app in foreground

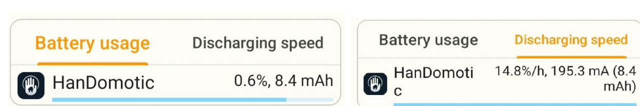

Figure 7: Consumption smartwatch app while performing SVM classification

The second set of images shows the results obtained with the smartwatch app (fig: 6, 7). When the smartwatch app is in foreground we can notice a bigger energy consumption with respect to the smartphone app. This is because, while in foreground, the

ServerSocket Thread of our smartwatch application is kept open and waiting for new settings that might arrive from the smartphone. However, this process is not necessary when the app is in background; in fact the ServerSocket gets shut down in that case and the energy consumption is significatively lower. During the gesture-dectection phase, as expected, we have an higher energy consumption level.

4 CONCLUSION

The HanDomotic project successfully proved the feasibility of using a gesture-based domotic control system leveraging Bluetooth Low Energy (BLE) beacons and wearable technology. By integrating an Android smartphone and a WearOS smartwatch, the system provides an efficient method for indoor localization and gesture recognition. The BLE beacons enable low-energy, high-accuracy room detection, while the smartwatch's accelerometer data, processed through a Support Vector Machine (SVM) classifier, allows a reliable gesture recognition.

Overall, the classifier exhibits strong performance, particularly in clearly defined gestures such as "Circle" and "No gesture" (i.e. the absence of gestures). Future work could focus on improving the detection accuracy of the "Clap" gesture, potentially by enhancing feature extraction methods or refining the training dataset to better capture the nuances of this specific gesture. This improvement would further improve the robustness and reliability of the gesture recognition system, contributing to a more seamless and intuitive user experience.

REFERENCES

[1] Lorenzo Porzi, Stefano Messelodi, Carla Mara Modena, and Elisa Ricci. 2013. A smart watch-based gesture recognition system for assisting people with visual impairments. In *Proceedings of the 3rd ACM International Workshop on Interactive Multimedia on Mobile & Portable Devices* (Barcelona, Spain) *(IMMPD '13)*. Association for Computing Machinery, New York, NY, USA, 19–24. https://doi.org/10.1145/2505483.2505487

[2] Mario Quinde, José Giménez-Manuel, Chimezie Leonard Oguego, and Juan Carlos Augusto. 2020. Achieving Multi-User Capabilities through an Indoor Positioning System based on BLE Beacons. In *2020 16th International Conference on Intelligent Environments (IE)*. 13–20. https://doi.org/10.1109/IE49459.2020.9155011