A Social IoT-based platform for the deployment of a smart parking solution

Published Version:
Floris A., Porcu S., Atzori L., Girau R. (2022). A Social IoT-based platform for the deployment of a smart parking solution. COMPUTER NETWORKS, 205, 1-12 [10.1016/j.comnet.2021.108756].

Availability:
This version is available at: https://hdl.handle.net/11585/862325 since: 2022-02-21

Published:
DOI: http://doi.org/10.1016/j.comnet.2021.108756

A Social IoT-based platform for the deployment of a Smart Parking solution[1]

Alessandro Floris[a,b], Simone Porcu[a,b], Luigi Atzori[a,b], Roberto Girau[c]

[a] *DIEE, University of Cagliari, 09123 Cagliari, Italy*
[b] *CNIT, University of Cagliari, 09123 Cagliari, Italy*
[c] *DISI, University of Bologna, 40126 Bologna, Italy*

Abstract

In this paper, we propose a novel IoT-based smart parking (SP) solution which has been designed to provide information on the status of parking spots offered in on-street parking areas. We mostly focused on the following issues of the state of the art solutions: scalability, interoperability to address the heterogeneity of IoT devices, low energy consumption, and timely prediction of the availability of the parking spots. To this we leverage the Social IoT (SIoT) Lysis environment to create virtual entities of the real world objects involved in the SP system for on-street parking areas. The usage of the social virtual entities allows for addressing the interoperability issues among different types of IoT devices used by separate solutions deployed in adjacent areas. Magnetometer sensors are used to automatically detect the presence of a vehicle in each parking spot and the sensors data are collected through concentrators that cover the whole parking areas through low-energy Wi-Fi. Additionally, a control dashboard has been designed and developed to manage the monitored parking areas and provide responsive data analytics regarding the occupancy of parking areas in the city, which can be accessed through an Android App. Finally, a smart payment service allows the users to automatically pay for the used services making use of Bluetooth beacons. Experiments have been performed with the developed test-bed to show the performance of the system to timely detect the presence

[1] A preliminary version of this paper was presented at the 2020 IEEE International Smart Cities Conference (ISC2), Piscataway, NJ, USA, 2020.

of a vehicle and identify the owner ID to trigger the payment procedure.
Keywords: Internet of Things, Vehicle detection, Smart Parking.

1. Introduction

Finding parking spots for private vehicles in public areas is a well-known issue whose importance increased in the last years due to the rapid increment of population living in urban environments. Studies show that, on average, 30 percent of the traffic is spent by drivers cruising for parking, and the average time to find a curb space ranges between 3.5 and 14 minutes [1]. The increment of traffic congestion due to the drivers looking for a free parking spot in the city has also a huge impact on the quality of life (QoL) of the citizens (e.g., frustration due to the waste of time, accidents) and, most importantly, on the environment, in terms of fuel consumption and air quality. For example, recent estimates observed up to 8.9 million excess deaths per year due to outdoor air pollution, which is in part caused by toxicity of vehicle exhaust [2].

Therefore, there are important reasons to study and implement novel solutions for Smart Mobility aimed to improve the QoL of the citizens. Besides strengthening the public transportation services to minimize the need for citizens to take their own cars, many solutions related to Smart Parking (SP) systems have been proposed in the last years, which are aimed to support the citizens in finding free sparking spots in a short timeframe by minimizing the fuel consumption and avoid to drive towards occupied parking areas [3].

Emerging ICT technologies, such as Internet of Things (IoT) and Cloud Computing, have given an important push to the implementation of SP solutions in the last years [4, 5]. With IoT-based solutions, different kinds of sensor can be used to detect the occupancy of a parking spot, such as magnetometers [6], infrared sensors [7], or ultrasonic sensors [8]. Alternative solutions rely on cameras and deep learning algorithms for image recognition [9, 10]. Wireless networks are then used to collect and transmit the data regarding the occupancy of the monitored parking spots to a cloud IoT platform, where data storage and data processing services are implemented. The elaborated information is then made available to administrators and citizens through end-user applications.

Most of the SP systems already proposed in the literature concern the off-street parking scenario, which focuses on the improvement of parking efficiency at closed parking lots. Being closed, these parking areas are generally easier to be monitored as it is sufficient to apply entrance counters to count the number of occupied spots and then to use cameras or sensors to detect the occupation of specific parking spots [11, 12]. This control facilitates the implementation of SP applications that allow the drivers to reserve a parking lot in advance for the needed time [13]. Also, common issues of on-street parking, such as environment conditions and vandalism, are avoided. On the other hand, the monitoring of on-street parking areas is more challenging and the literature available is more scarce. For this scenario, there are still several open research issues that need to be addressed, such as timely prediction of the availability of the on-street parking spots in the city, utilization of low-power consumption devices and communication protocols, provision of system scalability needed to cope with big data, provision of system interoperability to address the heterogeneity of IoT devices, and integration of Machine Learning (ML) to provide smart advanced functionalities and services to the citizens [4].

To deal with the aforementioned open research issues, in this paper we leverage the Social IoT (SIoT) paradigm to design a SIoT-based platform for the deployment of a full-stack SP system for on-street parking areas. Specifically, the proposed solution relies on the SIoT-based Lysis environment [14], which creates a virtual entity for each real world object, which then establishes social relationships with other peers on the basis of their activities and characteristics. In particular, the proposed SP system is easily expandable with different kinds of hardware devices (e.g., sensors, low-cost electronic boards), which can cooperate in the system's logic because of virtualization entities, which are able to build social relationships with each other with the aim to guarantee interoperability, flexibility, trustworthiness and privacy [15, 16]. Moreover, easy upgradability is achieved thanks to the complete virtualization of the physical devices. Furthermore, the fact that the virtual entities can create social relationships allows for high system scalability because of the creation of a distributed system based on a social network that can be explored through the social connections between the various virtual entities. Finally, virtual super entities support the implementation of data aggregation and data analysis operations, which can run ML-based algorithms to process the data acquired by the sensors and extract information useful for the SP management.

In particular, the following are the main functionalities and novelties provided by the proposed SP system:

- Vehicle detection and identification: we leverage the magnetometer sensor for the board aimed at automatic detection of a vehicle entering or leaving the parking spot. We conducted vehicle detection experiments with 15 different vehicles by considering three different magnetometer positions under the vehicle to find the best sensor placement achieving the greatest detection rate. To recognize the vehicle, we based on Bluetooth Beacons to uniquely identify different vehicles and their respective owners. Finally, social relationships among the parking spot's virtual entity, the car's virtual entity and the virtual entity of the driver's smartphone were leveraged to achieve timely prediction of availability of the free parking spots in the city.

- Data transmission: energy-saving routines were designed for the data transmission board, which performs the vehicle identification process and the data transmission only when notified by the vehicle detection board. Low-power Bluetooth and Wi-Fi modules were utilized. Concentrators covered the parking areas to transmit the collected data to the Cloud platform.

- Cloud platform: the SIoT-based Lysis environment enables for the creation and management of virtual entities that collect the data acquired by vehicle detection boards, and virtual super entities that perform data analysis tasks using ML-based algorithms. These virtual entities fully implement the SIoT paradigm and therefore can establish social relationships between each other by providing system interoperability, scalability, flexibility, trustworthiness, and privacy. We also demonstrate the potentialities of a Data Analysis service implemented in a super virtual entity, which predicts the occupancy state of the parking spots when trained on a parking dataset.

- Control dashboard: at the highest layer of the Cloud platform we developed a dashboard where administrators can manage the monitored parking areas and control data analytics and statistics regarding the occupancy of parking areas in the city. This high-level end-user application is totally based on the data collected and managed by the virtual entities and virtual super entities running on the Cloud platform.

- Android App: we developed an Android App where the citizens have a real-time view of the state of occupancy of parking areas in the city. This App communicates with the Cloud platform through ad hoc APIs to provide real-time updated information to the citizens.

- Smart payment service: is an e-wallet service that allows citizens to pay the actual parking time through the Android App. The actual parking time is measured thanks to the proposed vehicle detection and identification services.

The paper is structured as follows. Section 2 discusses the major related works in this area. Section 3 presents the proposed smart parking system. In Section 4, we discuss the results of the vehicle detection and identification processes as well as privacy and security issues. In Section 5, the Control Dashboard and the Android App are presented whereas Section 6 demonstrates the potentialities of the data analysis module implemented by the proposed platform. Finally, Section 7 concludes the paper.

2. Background

There are many studies proposing SP systems based on IoT in the literature. A comprehensive survey of SP solutions is provided in [3, 4, 5, 17, 18]. In particular, a classification of SP systems is proposed in [4, 5, 17]; with respect to this classification, our SP system belongs to the Parking Guidance and Information Systems (PGIS) category, whose main objective is to show and guide the user to available parking slots so as to reduce the time spent searching for free spaces. A PGI system typically consists of four main components: parking monitoring mechanism, parking space information dissemination, telecommunications network, and control center. In the following, we discuss for each of these components the limitations of state-of-the-art solutions and the novelties introduced by the proposed SP system, which is particularly focused on on-street parking areas. Table 1 summarizes some of the limitations of literature solutions.

2.1. Parking monitoring

The main objective of the parking monitoring component is to detect when a vehicle arrives at or leaves a parking slot. To this aim, many types of vehicle detection sensors can be used. Paidi et al. discuss the strengths and drawbacks of these sensors by considering the diverse characteristics of indoor and outdoor parking areas [13]. The authors identified some kinds of sensors sensitive to environment as most suitable for indoor parking areas (e.g., RFID, infrared and ultrasonic sensors) whereas others are suitable for both indoor and outdoor parking areas, such as magnetometer and microwave radar. For example, RFID readers are used in [7] to control the entry and exit gates of an indoor parking, while infrared sensors are used to detect the occupancy state of single parking spots. Two ultrasonic sensors are used in [8] to measure the free space in both the front and back sides of the vehicle. If the measured space is lower then a specific threshold, the spot is set as occupied otherwise is set as free. An additional drawback of these sensors is that they need a wall near the spot (or some structure above the spot) where to be placed to sense the presence of the car based on the change of measured energy [19, 20, 12]. The solutions based on this kind of sensors are then not suitable for on-street parking. Roman et al. [11] utilized an ultrasonic rangefinder installed on the passenger side of a vehicle, which served as the mobile sensing system to detect vehicles and spaces on the streets. Although this system can identify parked cars or empty spaces accurately, it requires a vehicle continuously cruising on the streets. Alternative to sensors, cameras are used to monitor on-street parking spots together with deep learning algorithms [9, 10]. Visual cameras can be used for vehicle detection and identification (e.g., license plate recognition [12]) using machine vision. However, these solutions need an open view and their performance can be significantly impaired by weather conditions [4]. Also, the cost of cameras is not negligible, in particular when compared with sensors. For these reasons, we decided to utilize the magnetometer sensor to detect the presence of vehicles, which is a cheap solution that can be used to monitor both indoor and outdoor parking areas. In [21], a sensor node with magnetometer sensor is placed between two adjacent parking spots on the ground to detect which spot is free/occupied. The vehicle detection rate of this module is said to be 98% but no details about the conducted experiments are provided. In [22], an array of 5 tri-axial magnetic sensors was placed on the roadside to detect the presence of a vehicle. The sensor placed in front of the single vehicle can clearly detect it both in the parking spot and during the parking approach. However, when in presence of other cars, a signal processing cleaning process was needed to be able to detect the vehicle. Another limit of this study is that only two cars were used for experiments. A vehicle detection algorithm based on the magnetic variations induced by a car while parking in a spot is provided in [23], which achieves an accuracy of about 98.5%. However, it is not clear if the experiments were performed with different vehicles or with the same vehicle. The magnetic signal is combined with the signatures of Ultra Wide Band channel in [6] to improve the accuracy of detection results. Authors found that the maximum magnetic strength appears in the vicinity of the front axles or rear axles of the measured vehicle. On the contrary, the magnitude of measured magnetic variation is very weak between these axles. The detection accuracy of this method is 98.81%. Even though the aforementioned studies have already presented vehicle detection systems based on a magnetometer, comprehensive results are not still clearly provided regarding the detection performance of this sensor. Therefore, to overcome the limitations of these studies, we conducted vehicle detection experiments with 15 different vehicles by considering three different magnetometer positions under the vehicle to find the best sensor placement that optimizes the vehicle detection rate. The results are discussed in Section 4.1.

2.2. Parking space information

This component aims to provide information regarding the availability of parking slots. Besides the different techniques that can be used to disseminate the information, such as mobile/web applications [24] and even SMS [25], the main open challenge is the timely prediction of the availability of the on-street parking spots in the city. As discussed in Section 4.3, the proposed SP system leverages social relationships, established among the virtual counterparts of the parking spot, the vehicle, and the user's smartphone, to identify possible free parking spots by detecting in advance when the driver is returning to his car and he is likely to free the spot. These possible free parking slots are then shown on the parking map of the mobile App so that the citizens looking for parking can drive in that direction minimizing the time spent cruising for parking. This technique is combined with the vehicle identification implemented at each parking spot through identification of Bluetooth Beacons on board of the vehicles used to uniquely identify the vehicle and the respective owner, as discussed in Section 4.2.

The flexibility of the proposed SP solution in providing real-time information regarding available parking spots in the city enhances the static reservation solutions offered by most of literature studies [19, 26]. Again, this kind of reservation-based solution is more suitable for controlled indoor parking areas and not on-street as it does not allow for a dynamic utilization of the parking areas in the city because reserved spots remain unused until the user who reserved the spot occupies that spot. Furthermore, in some cases the users need to physically check-in using short-range communications, such as NFC, or via traditional cash and credit cards [20, 26]. The count of the actual parking time is another limitation due to parking reservation [20, 26]. Indeed, this solution require the user to reserve in advance the amount of time during which the vehicle will be parked. However, it may happen that the actual parking time will result to be shorter (waste of money) or longer (requiring the user to remove the car in case the next hours are already reserved by another person) then the reserved one. Besides detecting the parked vehicle autonomously and counting the actual parking time, our SP system also enables the e-wallet payment via mobile App.

2.3. Telecommunications network

This component consists of the telecommunications network that enables the communication between all system layers. The main classification regards the utilization of long-range low power wide area networks (LPWAN) or short-range wireless networks. LPWAN can be used to cover wide ranges with low power consumption. For example, in [27], a Narrowband Internet of Things (NB-IoT) module is used to transmit the data acquired by the sensor node to the cloud server. The drawback of LPWAN standards is that they have not yet been widely adopted in many areas and regions and for this reason their application is limited [4]. On the other hand, short-range communication technologies such as Bluetooth, Wi-Fi, and ZigBee have been widely used and tested to establish short distances communication. For this reason, these standards are widely adopted for SP solutions. For instance, Bluetooth low energy (BLE) beacons are used in [28] to uniquely identify registered parking spots. However, users need to use the Bluetooth interface of their smartphone to scan the closest Beacon related to a free spot. The Wi-Fi is typically used in SP system to provide Internet connectivity to the sensing nodes to transmit the collected data to the cloud server [29, 30]. In the proposed SP system, we based on low-power Bluetooth to implement vehicle identification and low-power Wi-Fi to implement short-range communications between the electronic board at the parking spot and the Concentrator device, which, in turn, employs cellular communication technologies to transmit the data to the cloud platform.

2.4. Control center

The control center is the core of the SP system and aims at processing the data received from parking facilities and controlling the display of availability information to the users. Many SP systems from the literature based on the layered IoT architecture to propose their own control center. The architecture proposed by Rico et al. [26] is composed of 3 elements: Smart Server, Smart Object, and Smart Mobile. The Smart Object acts as a bridge between the vehicle detection sensors and the Smart Server and between the parking spot and the Smart Mobile (for the check-in). The Smart Server accounts for data collection and processing and provides the Smart Mobile with the parking status. Finally, the Smart Mobile enables the citizen to check for parking availability and to check-in through the Smart Object once the vehicle is parked. However, the implementation of the vehicle detection system is not discussed and users have to check-in with short-range communications. In [31], a fog computing architecture is proposed to reduce the communication latency between the sensor nodes and the control center. However, the authors provide a theoretical discussion with no implementation or experimental results. The UParking system is discussed in [32], which leverages the IoT paradigm to provide several functionalities for the citizen, such as, listing of available parking slots, booking, payment, statistics, and controlling parking access. However, the proposed implementation is only suitable for indoor parking, such as most of the solutions proposed in the literature. To the best of the authors' knowledge, the proposed solution is the first leveraging Social IoT to provide smart and advanced services for SP.

3. Smart Parking system

The proposed Smart Parking (SP) system has been developed with the objective to provide the following main functionalities:

Table 1: Limitations of state-of-the-art SP systems.

Limitation	References	Proposed SP solution
Vehicle detection system not implemented	[26]	Magnetometer-based vehicle detection system
Ultrasonic/infrared/light sensor	[19, 20, 12]	Magnetometer-based vehicle detection system
Costly cameras	[9, 10, 33, 34]	Magnetometer-based vehicle detection system
Need to reserve the PS	[19, 26, 33]	No need to reserve the PS
No count of the actual parking time	[19, 26, 27, 33]	Count of the actual parking time
Need to pay the reserved parking time	[19, 26, 33]	Payment of the actual parking time
Need to physically check-in	[20, 26]	Parking confirmation via mobile App
Payment not implemented	[12, 34]	E-wallet payment via mobile App
Mobile sensing approach	[11]	Fixed vehicle detection sensors

1. Autonomous and automatic timely detection and identification of the occupancy of the parking spots in a city in an on-road scenario.

2. Autonomous measurement of the actual parking time of a vehicle.

3. Automatic payment of the actual parking time through a smart payment service.

4. Real-time mapping of the state of occupancy of the parking areas in a city.

5. Control Dashboard for administrators and Smart Parking App for the citizens.

6. Data analytics and statistics regarding the occupancy of parking areas in a city.

The architecture of the proposed SP system is shown in Fig. 1 and it is based on the Lysis architecture [14] as the IoT environment since it provides some advantages, such as high scalability due to the exploitation of the social IoT paradigm, high flexibility and easy upgradability thanks to the complete virtualization of the physical devices. The proposed SP system includes all levels of common SP architectures [4]. In particular, the architecture is divided in 4 levels: the Hardware level includes the physical sensing devices that acquire data from the real world for detecting the presence/absence of the vehicles. It also

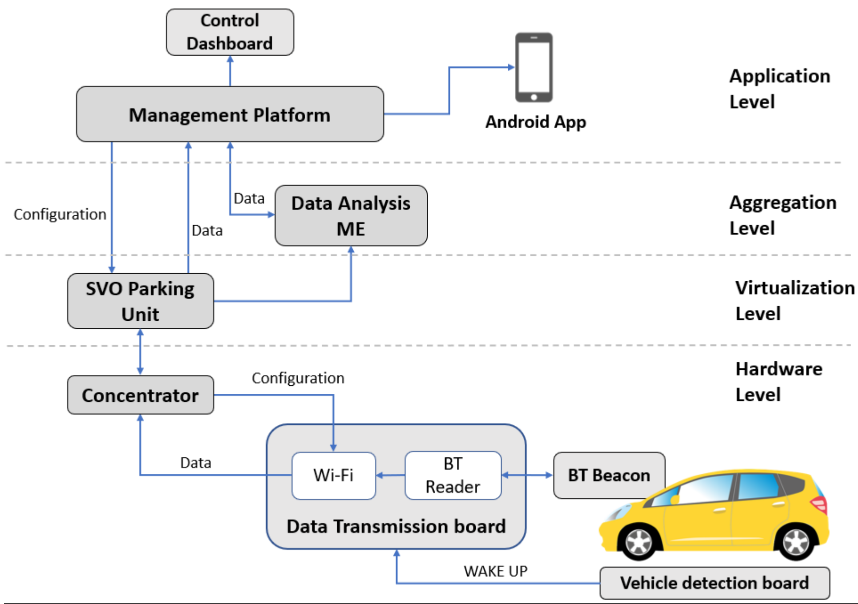


Figure 1: Architecture of the proposed Smart Parking system.

includes the transmission of the data through the network towards the cloud platform. The Virtualization and Aggregation layers act as the middleware, where the data is stored and processed to be suitable for the upper layer, i.e., the Application layer. At this level, software platforms and services are developed in the cloud, which allow to manage and visualize the data as well as to interface the smart parking services with the administrators and the end-users.

In the following we provide a detailed description of each level of the architecture.

3.1. Hardware level

The Hardware level includes all sensors and devices that acquire information regarding the occupancy of the parking spots as well as the data transmission devices. In particular, the following physical devices belong to the Hardware level:

- Vehicle detection board: detects the presence of a vehicle parked in the spot to determine whether the spot is free or occupied. It is composed of a magnetometer sensor to detect the presence of the vehicle and of an electronic board for data processing and transmission. When a change on the state of the parking spot is detected, a Wake up signal (interrupt) is sent to the Data transmission board.

- Bluetooth (BT) Beacon: is used to uniquely identify the vehicle. It has to be placed into the vehicle.

- Data transmission board: is an electronic board that has to be equipped with two wireless interfaces (BT and Wi-Fi). When the Vehicle detection board detects the presence of a vehicle in the spot, it sends a Wake up signal to the Data transmission board, which activates the BT interface to scan the BT devices in the proximity and identify the BT Beacon of the vehicle that occupies the parking spot. When the data is collected, the Data transmission board activates the Wi-Fi interface to transmit the data to the Concentrator device.

- Concentrator: makes use of short-range wireless communication technology (i.e., Wi-Fi) to collect the data from all Data transmission boards installed in a parking area. Cellular communication technology (i.e., 4G) is then used to transmit the collected data to the Virtualization layer of the cloud platform.

3.2. Virtualization level

At the Virtualization level, a Social Virtual Object (SVO) Parking Unit is created for each parking spot that must be monitored. Each SVO is an autonomous web service that acts as a virtualization of the physical device with which it interfaces directly with REST APIs and represents it with all its features and functionalities. The SVO Parking Unit receives from the Data transmission boards installed in the parking areas the information regarding: vehicle ID (BT beacon ID), parking spot ID, parking spot state (free, occupied, or possible free), state of charge of the battery of the Data transmission board, and timestamp. The SVO stores this data and make them available to the upper levels.

This level fully implements the SIoT paradigm through device virtualization. The fact that the SVOs can create social relationships allows us to create a distributed system based on a social network that can be explored through the social connections between the various SVOs. A SIoT social network consists of well-defined types of relationship between objects. If the objects belong to the same owner they can create an Ownership Object Relationship (OOR). A Co-location Object Relationship (CLOR) can be formed if the objects are fixed and positioned in the same location. Objects of the same model or production line can create a relationship called Parental Object Relationship (POR). Objects that meet in their owners' workplaces and that can collaborate in this context can create a Co-work Object Relationship (CWOR). Finally, a Social Object Relationship (SOR) is formed as a result of numerous encounters between things, such as those that occur between customers' phones and the shop's equipment (scales, barcode readers, etc.) [15]. The relations are then exploited to search for data for statistical analysis on parking areas. Using virtual counterparts allows us to implement much of the logic outside of the physical devices so to save on battery. It also allows the easy integration of other elements in the architecture, which are therefore able to connect through other social relationships and immediately usable by the higher levels thanks to the hardware abstraction. We leverage social relationships to identify possible free parking spots by detecting in advance when the driver is returning to his car and he is likely to free the spot (see Section 4.3). The fact that all SVOs expose uniform APIs to higher levels makes it possible to modify the application that uses their data or add new ones without having to change the software on the devices.

3.3. Aggregation level

At the Aggregation level, super entities called Micro Engines (MEs) are implemented, which are a composition of more than one SVO. The ME has the capability to inherit and increase the functionalities and capabilities of the SVOs of which it is composed. Furthermore, ML algorithms can be implemented in MEs to perform data aggregation and data processing operations on of SVOs' data. An example of operation that can be performed by MEs is the provision of statistical analysis of aggregated data collected by SVOs regarding the parking spot occupation of a specific parking area. This information may concern, for example, the average number of free/occupied parking spots in the monitored parking areas of the city, average parking time, the most used parking areas, etc. The analysis of this data can be utilised by the city administrators to improvement the parking areas and services of the city. In Section 6, we present a practical example of the potentialities of the Data Analysis ME.

3.4. Application level

Finally, at the Application level, the Management Platform supports the Control Dashboard for the management of the SP system and the Android App developed for the citizens. The Management Platform includes all back-end services needed to manage the data collected by the lower layers, e.g., the database, whereas the Control Dashboard acts as the front-end application for administrators for managing the SP service.

The Control Dashboard allows to add different parking areas of the city to be monitored and provides a real-time mapping of the occupied parking spots. Also, thanks to the monitored data and citizen registrations, it provides further services, such as the ability to classify the parking spots (for example, those reserved to disabled or resident) and then to establish the occupation of an unauthorized spot and reporting to the competent authorities (e.g., the occupation of a spot for disabled/residents by a non-disabled/non-resident citizen).

With regard to the support of the Android App, the Management Platform performs the following actions: sends the parking notification which confirms to the citizen that its parking has been registered; computes the actual parking time once the occupant vehicle has released the spot; sends the payment notification by automatically calculating the parking cost based on the actual parking time and the rates applied for that parking area. Besides these functions, with the Android App the citizens have the possibility to visualize a real-time mapping of the occupancy of city parking areas, so as they can know where it is more likely to find free parking, avoiding having to go around empty in the city and therefore wasting fuel with a consequent reduction of pollution and of wasted time. Furthermore, through the Android App the citizens have the possibility of associating an electronic payment account so that the cost of parking is deducted without the need for further payment actions.

In Section 5, we present the details of the Control Dashboard and the Android App.

4. Vehicle detection and identification

In this section, we present the experimental results obtained with the proposed Vehicle detection board (Section 4.1), we describe the vehicle identification process (Section 4.2), we leverage social relationships to identify possible free parking space in advance (Section 4.3), we discuss privacy and security issues (Section 4.4), and we estimate implementation costs (section 4.5).

4.1. Vehicle detection

The Vehicle detection board has the task to detect the presence of the vehicle in the parking spot and then determine its occupancy state. For the implementation, we used the tri-axial magnetometer sensor (QMC5883L) as the vehicle detection sensor and the Arduino Pro Mini as the board to collect and process the sensed data. The magnetometer is a sensor that measures the changes of the geomagnetic strengths caused by the presence of ferromagnetic metal objects, such as the vehicles. The magnetometer outputs are the magnetic strengths measured in x-, y-, and z-direction (G_x, G_y, and G_z, respectively) and the magnetic vector magnitude G (we will refer to it only as the magnitude in the following), which is computed as

$$G = \sqrt{G_x^2 + G_y^2 + G_z^2}. \tag{1}$$

From preliminary experiments, we found that the measured magnetic strengths and magnitude are influenced by the bearing, because of the influence of geomagnetic strengths. Therefore, we decided to place each magnetometer oriented to the North in order to set the same bearing reference for each measurement and avoid bias due to the bearing influence. Also, we noted that the magnetic strengths are quite variable over the time whereas the magnitude is more stable. Therefore, we decided to only consider the magnitude measures for vehicle detection experiments.

Moreover, we investigated the influence of the position of the magnetometer sensor under the vehicle because the distribution of ferromagnetic materials in the vehicle induces magnetic variations [6]. For this reason, we considered three different positions under the vehicle where to place the magnetometer, namely, sensor under the front axle (P1), sensor under the center of the vehicle (P2), and sensor under the rear axle (P3). As an example, we show in Fig. 2 the variations of the measured magnitude for a Nissan Micra as a function of the sensor position. When the parking spot is free, the magnitude value is about 21 μT. When the vehicle is parked, the measured magnitude is always greater than 21 μT. In particular, magnitude values of about 39 μT, 25 μT, and 29 μT are measured when the sensor is place in position P1, P2, and P3, respectively. This difference between the magnitude measured when the parking spot is free and when the vehicle is parked can be used to determine the presence of the vehicle in the parking spot. Clearly, the greater this difference the greater the reliability of the vehicle detection. For instance, for the specific case of the Nissan Micra in Fig. 2, the best choice would be to place the sensor under the front axle of the vehicle as the measured magnitude is doubled when compared to the magnitude measured when the spot is free.

However, different vehicles have different distribution of ferromagnetic materials and can induce different magnetic variations. Therefore, we extended the study regarding the vehicle detection and the optimal position of the sensor by considering 12 different vehicles. In the same parking spot we placed a magnetometer sensor and we measured the magnitude when the parking spot was free

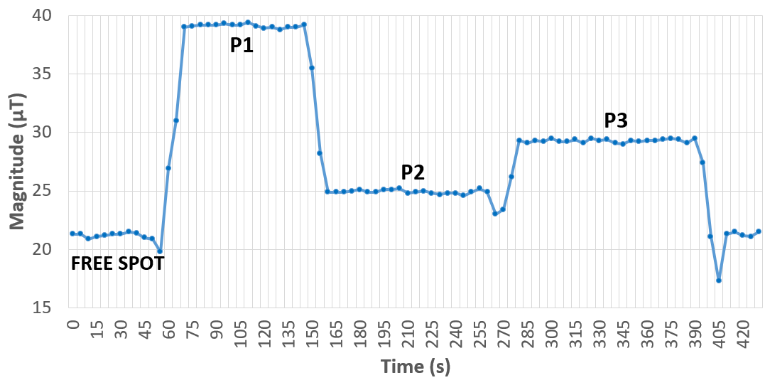


Figure 2: Magnitude measured for a Nissan Micra with the sensor placed in three different positions under the vehicle. P1: sensor under the front axle; P2: sensor under the center of the vehicle; P3: sensor under the rear axle.

Table 2: Magnitude (μT) measured for 12 different vehicles and 3 sensor positions. PS: parking spot.

Vehicle	**PS free (Reference)**	**PS occupied**			**Magnitude variance**		
		P1	**P2**	**P3**	**P1**	**P2**	**P3**
Toyota Aygo	21.77	45.98	47.86	27.98	24.21	26.09	6.21
Nissan Micra	21.21	39.11	24.92	29.29	17.90	3.71	8.08
Citroen C3	21.71	25.05	34.48	24.82	**3.33**	12.76	3.11
FIAT Grande Punto	21.79	41.29	20.03	33.98	19.50	-1.76	12.19
Ford Focus	21.64	33.98	21.30	33.67	12.34	**-0.34**	12.03
Alfa Romeo 147 serie 1	21.85	31.40	25.65	24.13	9.56	3.80	**2.28**
Peugeot 308	21.66	121.29	35.23	28.93	99.64	13.58	7.27
Alfa Romeo 147 serie 2	21.68	31.12	25.34	15.48	9.44	3.66	-6.20
KIA RIO	22.28	11.40	36.89	43.32	-10.88	14.61	21.04
Citroen C2	21.36	29.70	37.70	30.44	8.34	16.33	9.08
Lancia Ypsilon	21.68	29.90	35.26	31.61	8.23	13.58	9.94
Citroen C1	21.98	47.02	17.53	31.91	25.04	-4.44	9.93

(Reference value) and when the vehicle was parked. Also, the magnitude was measured with the magnetometer sensor placed in the 3 different positions P1, P2, and P3.

The collected magnitude values measured for the 12 different vehicles are summarized in Table 2. Each value in the table is the mean value of about 36 measures since the Vehicle detection board was set to acquire data with a time step of 5 s for a total of 3 minutes. First, it can be seen as the reference magnitude value measured when the parking spot is free is quite the same before parking each vehicle. The total magnitude mean (computed considering all measurements) is 21.72 μT with a standard deviation of 0.26 μT. In the middle part of the table (PS occupied) the magnitude values measured when the parking spot is occupied by the vehicle for the 3 sensor positions are shown. It can be observed as different vehicles cause very different variations in the measured magnitude, even when the sensor is placed in the same position. In general, the magnitude measured when the car is parked is greater than the Reference value. However, this is not always true (5 times out of 36: 1 for P1, 3 for P2 and 1 for P3). The right part of the table (Magnitude variance) shows the difference between the magnitude measured in the absence and in the presence of the vehicle for the 3 considered sensor positions. The smallest values are highlighted in bold for each of the 3 sensor positions, i.e., 3.33 μT for P1 (Citroen C3), -0.34 μT for P2 (Ford Focus), and 2.28 μT for P3 (Alfa Romeo 147 serie 1).

From the collected magnitude results, we can conclude that the optimal position for the magnetometer sensor for the detection of the parked vehicle is under the front or the rear axle of the vehicle. In particular, by setting a threshold of 2 μT, the Vehicle detection board would always be able to detect all vehicles considered in this study. We do not consider the position under the center of the vehicle because the minimum measured magnitude variance is too small (-0.34 μT) to be reliably detected. For validation tests, the sensor was then placed at about one meter of distance from the start of the parking spot. In this way, when the vehicle parks on the spot the sensor will be under the rear or the front axle, depending on the parking direction decided by the driver. For validation tests, we parked 3 vehicles different from those used to collect the data in Table 2, i.e., a Ford Kuga, a Renault Clio and a Renault Zoe. All the 3 vehicles were detected by the Vehicle detection board for both the parking directions, meaning that the measured magnitude difference was greater than the set threshold, which then results reliable for vehicle detection.

We have also conducted a second experiment to investigate whether it is possible to detect, with a single sensor, the occupancy of the 2 parking spots adjacent to the parking spot where the magnetometer is placed. To this, we measured the magnitude when the vehicle is parked in the central spot (where the magnetometer is placed) and the magnitude when a first and then a second vehicle are parked in the adjacent spots. However, the measured magnitude values did not provide relevant variations that may be helpful in the detection of the occupancy of the adjacent spots. Therefore, for the proposed SP system each parking spot must be equipped with a Vehicle detection board.

4.2. Vehicle identification

When the Vehicle detection board detects the presence of a vehicle in the monitored parking spot, it sends a Wake up signal to the Data transmission board. We implemented the Data transmission board with the Telit cs2060a device, which is a battery-powered module equipped with low-power BT and Wi-Fi interfaces. This board then performs the scan of the BT devices detected in the proximity of the board with the objective to identify a BT beacon whose ID is registered to the SP service. We used the BlueUp BlueBeacon Mini as the BT beacons in this case. The first part of the read IDs is identical for each BT beacon used for the SP service so that the board is able to recognize them and discard other BT devices. The last part of the IDs is different for each user and identifies them (and their vehicle).

If just one BT beacon is found after the scan, the following data is immediately sent to the Concentrator (implemented with the Telit cs2091 device): BT beacon ID (vehicle/user ID), parking spot ID, parking spot state (occupied), state of charge of the battery of the Telit device, and timestamp. When the parking spot is released, the same data is sent but with the parking spot state set to free. If more than one BT beacons are found, e.g., in case more vehicles park at the same time in adjacent spots, the Data transmission board selects the 3 BT beacons detected with the greatest signal power (RSSI). Indeed, we have found from experimental results that each Vehicle detection board can cover up to 3 adjacent parking spots by placing it at the central spot. When the user parks the car and the SP system detects the parked car and identifies the vehicle/user, a notification is sent to the App of the user confirming that the car is parked. The count of the parking time is then started. When the parking slot is released, the SP system sends a notification to the App of the user containing the actual parking time and the parking tariff to be paid.

Finally, if a parked vehicle is detected but no BT beacon is found by the board, the parking spot is set as occupied by the system (and it also appears as occupied in the map). Indeed, it is assumed that also citizens that are not registered to the proposed SP service can park in the monitored parking area, by paying the parking with traditional services.

4.3. Notices on availability of free parking spaces

The proposed system provides for the signaling of vehicles that are about to leave the parking lot thanks to the social interaction of the SVOs.

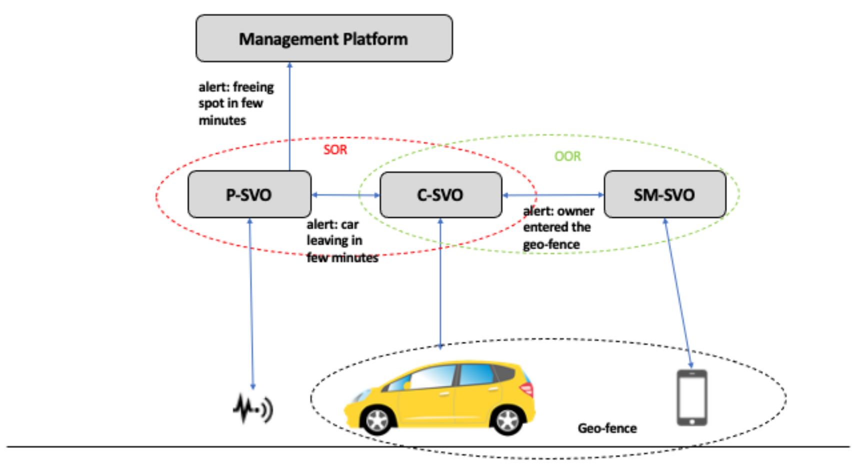


Figure 3: The alerting process through the social relations of the SVOs.

As shown in the figure, the parking SVO (P-SVO) creates a SOR relationship with the car's SVO (C-SVO) after five minutes of presence. The latter has an OOR relationship with the driver's smartphone SVO (SM-SVO) since they belong to the same user. When the car parks, the C-SVO sends a message to the SM-SVO with the parking position so that a control is generated on a geo fence around the parking lot. When the driver is returning to his car, upon entering the geo fence the SM-SVO sends an alert to the C-SVO, which forwards the message to the P-SVO. The latter sends a message to the platform to report a possible free parking. The car park is thus marked in orange on the map so that those looking for parking can drive in that direction. This functionality, totally based on the social relationships established among the SVOs of the proposed SP system, enables to timely predict the availability of free parking spots in the city. As a consequence, citizens are promptly notified about these possible free spots, which allows them to minimize the time spent cruising for parking.

4.4. Privacy and security

With the proposed SP system, the privacy and security of the citizens is preserved. The user has to register with the SP service provider before being able to utilize the SP service. Once registered, the user is provided with a BT beacon, whose ID is associated with the user in the database of the cloud platform. Therefore, when the vehicle of the user is detected in the parking spot, only the ID of the BT beacon is read by the Vehicle detection board and sent to the cloud platform. No personal user information is exchanged. In the platform, the software checks whether the received ID beacon is associated to a registered user. In the positive case, a verification message is sent to the App of the user, where the user is asked to confirm whether his vehicle has been parked in that moment. If the user confirms, the parking time count starts.

This process allows to avoid attackers to park their car in the parking spot and then simulate the presence of a BT beacon with the ID of a BT beacon owned by a registered user and previously obtained by means of sniffing and spoofing attacks. Indeed, it is well known that BT beacons broadcast their data and whoever can read it. But the control procedure at the cloud platform and the confirmation request needed from the registered user overcome this problem. Moreover, the communications between the Concentrator and the cloud platform and between the cloud platform and the Android App are encrypted, which makes it harder for attackers to capture the data.

4.5. Implementation costs

In this section, we provide an estimation of the costs needed to implement the proposed solution for a parking area composed of 100 parking spots. Each parking spot needs an Arduino Pro Mini (5 $) and a QMC5883L magnetometer sensor (1.5 $). A Telit cs2060a (2 $) is needed to collect data from 3 parking spots. Therefore, 33 of these devices are needed for 100 parking spots. Finally, it is reasonable to assume that two Telit cs2091 (5 $) devices cover adequately the total parking area. However, the coverage of the Concentrator devices strictly depends on the spatial distribution of the parking spots and may vary. The total cost is about 725 $. However, this cost can be lowered buying significant quantities of devices that reduce the cost for equipping each single spot. A realistic cost estimation of the implementation costs for 100 parking spots can then hover around 500 $. Costs of the cloud services need to be considered as well; on the basis of the traffic generated and the amount of data that need to be stored for this size of parking area covered, the relevant costs are around 150 $ per month.

The cost of the BlueUp BlueBeacon Mini is about 10 $ and is included in the service cost paid by the citizen to benefit the SP service.

5. Dashboard and Android App

In this section, we present the Control Dashboard (Section 5.1) and the Android App (Section 5.2).

5.1. Control Dashboard

The Control Dashboard has been implemented for administrators for the management of the SP system as it allows to add and manage different parking areas of the city to be monitored and provides a real-time mapping of the occupied parking spots. The Control Dashboard is developed in Python Flask and it is hosted on the Google Cloud. Concerning the Views, the dashboard uses Jinja2, JavaScript, HTML and CSS to connect the back-end with the front-end and to display the data. The dashboard follows the rules given by the Model View Control (MVC) pattern.

The Views are structured as follows:

- Parking Areas: it shows the list of the parking areas already added to the system, as illustrated in Fig. 4, and allows the administrators to manage the existing parking areas and to add new ones. Indeed, the sub-menu includes the "Parking Management" section, the "Add Area" section and the "Parking Areas List". By clicking on the specific parking area tab, it is possible to display the details of the free/occupied/possible free parking lots, their position in terms of latitude and longitude, and the map of the parking area. An example is shown in Fig. 5.
- Users: it shows the list of all users subscribed to the SP service. This section allows the administrators to add and manage the users.
- Rates: it shows the list of all parking areas rates. This section allows the administrators to add and manage the parking tariffs.
- Analytics: it shows the analytics of all parking areas. For example, Fig. 6 shows the occupancy rate of the 2 parking areas in the system, namely, Ingegneria and Telit, in March.

5.2. Android App

The Android App has been developed for the citizens for three main objectives: allow controlling in real-time the occupancy state of the monitored parking areas; receive the notification of the parked vehicle autonomously detected

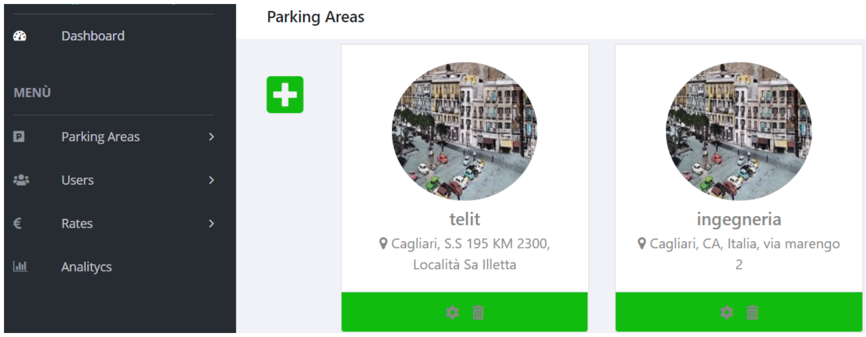
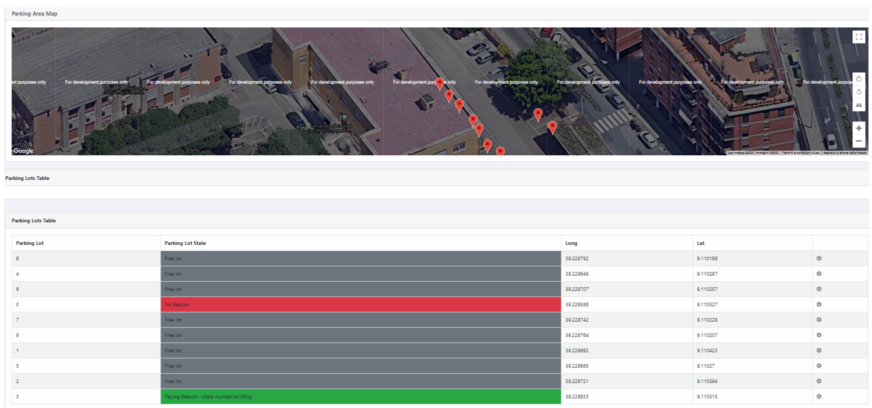


Figure 4: Control Dashboard: parking areas.



Parking Lot	Parking Lot State	Long	Lat	
9	Free lot	39.228792	9.110188	
4	Free lot	39.228649	9.110287	
6	Free lot	39.228707	9.110257	
0	No Beacon	39.228586	9.110327	
7	Free lot	39.228742	9.110228	
8	Free lot	39.228764	9.110207	
1	Free lot	39.228692	9.110423	
5	Free lot	39.228685	9.11027	
2	Free lot	39.228721	9.110394	
3	Paying Beacon - plate number:bx180cy	39.228633	9.110315	

Figure 5: Control Dashboard: details of the Ingegneria parking area.

by the SP system; receive the payment notification and perform the payment operation. Due to the very early stage of the system development, the first version of the smartphone App has been released only for Android-based devices. The used SDK is optimized for Android versions 11 and 10. Anyway, the older versions will be supported in the future new versions of the application.

The citizen who wants to utilize the SP system has to install the App and to register to the service. There are two ways to access the SP service. The first one is provided by the seller, i.e., the service provider. In this case, the citizen does the first access using its email and the Beacon ID (provided by the seller). After the first log-in step, the user can insert its private password, which becomes the only way to access its application profile. The second way

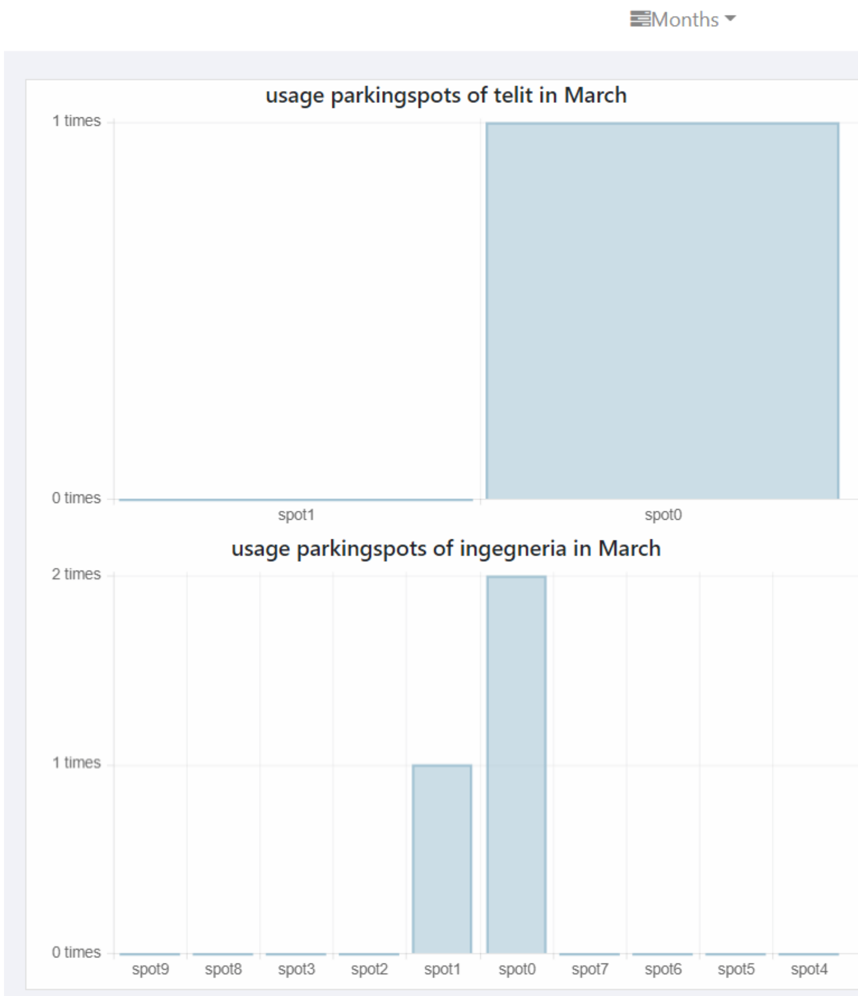
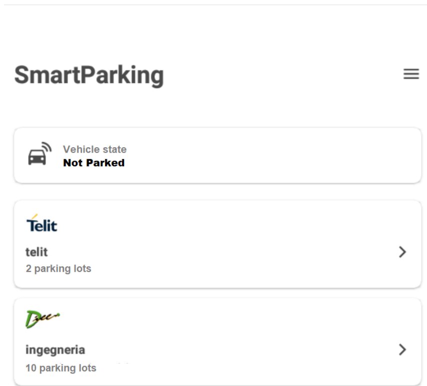
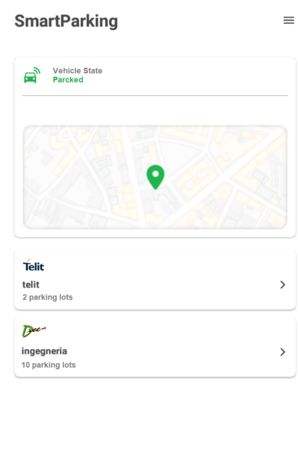


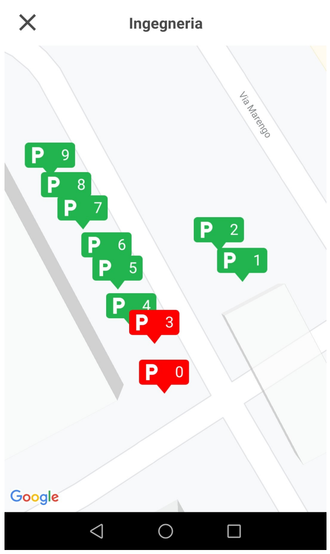
Figure 6: Control Dashboard - Analytics: occupancy rate of the Ingegneria and Telit parking areas in March.

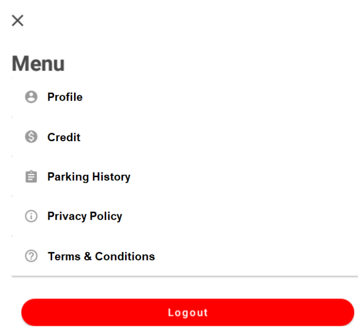




(a) Home screen: list of the parking areas and status of the vehicle (not parked).

(b) Home screen: list of the parking areas and status of the vehicle (parked).





(c) Parking screen: map of free/occupied parking lots at the Engineering faculty.

(d) Menu screen: profile and payment information.

Figure 7: Android App: Home and Menu screens.

to access the service is by registering from the application registration form. Anyway, the SP service requires its private BT beacon device, so the user will be unable to use the service until receiving the personal BT beacon, whose ID must be inserted into the profile information.

After the log-in, the users can check their credit, the history of the parking, and the profile info via the application's menu (Fig. 7(d)). The main window shows the vehicle state, such as parked or not. When the vehicle is parked, its position is also shown on the map (Figs. 7(a) and 7(b)). The list of the monitored parking areas is shown according to the user's position acquired from the device's GPS. By opening one parking area tab, the position and the occupancy state of each parking lot is shown, as in Fig. 7(c).

6. Data Analysis ME

In Section 3, we presented the architecture of the proposed SP system. At the Aggregation level, virtual super entities, called Micro Engines (MEs), can perform aggregation and processing functions of the data stored by the SVOs. In particular, we implemented a Data Analysis ME, which is aimed at providing statistical analysis and prediction models regarding the occupancy of the monitored parking areas in the city.

To demonstrate the potentialities of the proposed Data Analysis ME, we used the CNR parking dataset provided in [35] to train ML-based models aimed at predicting the occupancy state of the parking spots. Indeed, due to the difficulties in obtaining the authorisation for installing our boards in a city parking place, the preliminary data we have collected from the monitored parking areas were not sufficient to train a prediction model. The CNR dataset is collected in 23 days and contains occupancy information related to a parking area of 164 parking spots. The parking area is divided, in turn, into different 11 sections. In particular, the occupancy state (free or occupied) of the parking spots is known for each of the monitored day and hour.

There are several studies from the literature proposing models for predicting

Prediction outcome

		F	O
Actual value	**F**	86.7%	13.3%
	O	4.3%	95.7%

Table 3: Confusion matrix for the decision tree prediction model. F: free parking spot. O: occupied parking spot.

parking space availability. For example, the study in [36] highlights that a low complexity model, such as the decision tree model, can obtain better results than a Multilayer Perceptron (MLP) neural network for predicting the parking occupancy state during the day. However, a particular kind of neural network, i.e., the Recurrent Neural Network (RNN), can also be used to predict the parking occupancy state during the day. For instance, in [37], the authors used a Long-Short Term Memory (LSTM) RNN to forecast the occupancy rate of certain regions of the parking area.

For these reasons, we implemented two different prediction models: a decision tree model and an LSTM RNN model. The aim of these prediction models is to predict the occupancy state of a specific parking spot based on the particular day and hour. The decision tree model was trained with the maximum deviance reduction split criterion and the maximum number of splits of 1982. The weekday and hour data were set as the features, and the dataset was split with a 70%/30% training/validation combination and 5-fold cross-validation to avoid overfitting. The tree prediction model obtained a mean accuracy of 91.6%. The LSTM RNN comprises one sequence input layer, 1 LSTM layer, one fully connected layer, and the softmax layer. Also, the neural network used

	Prediction outcome F	Prediction outcome O
Actual value F	98.9%	1.1%
Actual value O	6.4%	93.6%

Table 4: Confusion matrix for the LSTM RNN prediction model. F: free parking spot. O: occupied parking spot.

the Adam optimiser and was trained for 100 epochs. The 5-fold-cross-validation was adopted to avoid overfitting. The LSTM RNN was trained with a different number of hidden units, respectively 50, 100, 150, and 200. A mean accuracy of 96.69% was obtained training with 100 hidden units. Indeed, the neural network was affected by overfitting training with 150 and 200 hidden units, whereas with only 50 hidden units it could not reach relevant results.

Table 3 shows the confusion matrix for the decision tree prediction model. It can be seen that when an occupied parking spot is predicted, the True Positive Rate (TPR) is 95.7% and the False Negative Rate (FNR) is 4.3%. On the other hand, when a free parking spot is predicted, the TPR is 86.7% whereas the FNR is 13.3%. Therefore, the prediction model is more accurate when it predicts that the parking spot is occupied while the prediction of a free parking spot is more uncertain. This is reasonable as the monitored parking spots were generally more occupied than free. However, the prediction value is almost 90% so it is an important information for the driver who wants to have some information regarding the possibility to find a free parking spot in that parking area for a specific day and hour. Table 4 shows the confusion matrix for the LSTM RNN, whose performance outperform those obtained by the decision tree model. Indeed, the LSTM achieves TPR values greater than 90% for both cases, i.e., 98.9% when a free parking slot is predicted and 93.6% when an occupied parking slot is predicted. Therefore, the LSTM is more accurate in predicting the free parking slots than the decision tree model, which makes the LSTM more suitable and reliable for a real SP scenario.

7. Conclusions

In this paper, we have presented a novel solution for smart parking which exploits a four-layer SIoT platform that enables to create virtual entities that can establish social relationships among each other to provide advanced functionalities for the SP system. The major features are: the automatic detection of the parking spot status; the timely prediction of the availability of free parking spots in the city; the automatic payment of the parking service by identifying each vehicle through Bluetooth beacons; the possibility to implement super virtual entities that can run ML-based algorithm and perform analysis such as models for predicting the occupancy status of parking lots.

Experimental activity has focused on the analysis of the magnetometer capability to detect the presence of the vehicles changing the vehicles characteristics and the position of the sensor under the vehicle. From the collected results, we concluded that the optimal position for the magnetometer sensor for the detection of the parked vehicle is under the front or the rear axle of the vehicle. With this position, by setting a threshold of 2 μT, the vehicle detection board would always be able to detect all the types of vehicles considered in this study.

Finally, to demonstrate the potentialities of the proposed Data Analysis ME, we have trained a tree decision model and an LSTM RNN to predict the status of the parking lots in a given area based on past data. The LSTM achieved an average accuracy of 96.69%, outperforming the performance of the decision tree (91.61%). Moreover, the LSTM resulted more accurate in predicting the free parking slots than the decision tree model (98.9% vs. 86.7%), which makes the LSTM more suitable and reliable for a real SP scenario.

Acknowledgements

This work has been partially funded by the Italian Ministry for the Economic Development (MISE), under the framework "Asse II del programma di supporto tecnologie emergenti (FSC 2014-2020)", project Monifive and by Italian Ministry of University and Research (MIUR), within the Smart Cities framework (Project Netergit, ID: PON04a200490).

References

[1] D. C. Shoup, Cruising for parking, Transport Policy 13 (6) (2006) 479–486. doi:https://doi.org/10.1016/j.tranpol.2006.05.005.

[2] M. R. Miller, D. E. Newby, Air pollution and cardiovascular disease: car sick, Cardiovascular Research 116 (2) (2019) 279–294.

[3] T. Lin, H. Rivano, F. Le Mouël, A Survey of Smart Parking Solutions, IEEE Trans. on Intelligent Transportation Systems 18 (12) (2017) 3229–3253.

[4] F. Al-Turjman, A. Malekloo, Smart parking in IoT-enabled cities: A survey, Sustainable Cities and Society 49 (2019) 101608.

[5] M. G. Diaz Ogás, R. Fabregat, S. Aciar, Survey of Smart Parking Systems, Applied Sciences 10 (11) (2020). doi:10.3390/app10113872.

[6] L. Lou, Q. Li, Z. Zhang, R. Yang, W. He, An IoT-Driven Vehicle Detection Method Based on Multisource Data Fusion Technology for Smart Parking Management System, IEEE Internet of Things Journal 7 (11) (2020) 11020–11029.

[7] P. Melnyk, S. Djahel, F. Nait-Abdesselam, Towards a Smart Parking Management System for Smart Cities, in: 2019 IEEE Int. Smart Cities Conference (ISC2), 2019, pp. 542–546.

[8] Y. KC, C. Kang, A Connected Car-based Parking Location Service System, in: 2019 IEEE Int. Conf. on Internet of Things and Intelligence System (IoTaIS), 2019, pp. 167–171.

[9] S. Gören, D. F. Öncevarlk, K. D. Yldz, T. Z. Hakyemez, On-Street Parking Spot Detection for Smart Cities, in: 2019 IEEE Int. Smart Cities Conference (ISC2), 2019, pp. 292–295.

[10] D. F. Öncevarlk, K. D. Yldz, S. Gören, Deep Learning Based On-Street Parking Spot Detection for Smart Cities, in: 2019 4th Int. Conf. on Computer Science and Engineering (UBMK), 2019, pp. 177–182.

[11] C. Roman, R. Liao, P. Ball, S. Ou, M. de Heaver, Detecting On-Street Parking Spaces in Smart Cities: Performance Evaluation of Fixed and Mobile Sensing Systems, IEEE Trans. on Intelligent Transportation Systems 19 (7) (2018) 2234–2245.

[12] R. Lookmuang, K. Nambut, S. Usanavasin, Smart parking using IoT technology, in: 2018 5th International Conference on Business and Industrial Research (ICBIR), 2018, pp. 1–6.

[13] V. Paidi, H. Fleyeh, J. Håkansson, R. G. Nyberg, Smart parking sensors, technologies and applications for open parking lots: a review, IET Intelligent Transport Systems 12 (8) (2018) 735–741.

[14] R. Girau, S. Martis, L. Atzori, Lysis: A platform for IoT distributed applications over socially connected objects, IEEE Internet of Things Journal 4 (1) (2016) 40–51.

[15] L. Atzori, A. Iera, G. Morabito, SIoT: Giving a Social Structure to the Internet of Things, IEEE Communications Letters 15 (11) (2011) 1193–1195. doi:10.1109/LCOMM.2011.090911.111340.

[16] V. Loscri, P. Manzoni, M. Nitti, G. Ruggeri, A. M. Vegni, A Social Internet of Vehicles Sharing SIoT Relationships, in: Proceedings of the ACM MobiHoc Workshop on Pervasive Systems in the IoT Era, PERSIST-IoT '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 1–6. doi:10.1145/3331052.3332470.

[17] A. O. Kotb, Y.-c. Shen, Y. Huang, Smart Parking Guidance, Monitoring and Reservations: A Review, IEEE Intelligent Transportation Systems Magazine 9 (2) (2017) 6–16. doi:10.1109/MITS.2017.2666586.

[18] A. Fahim, M. Hasan, M. A. Chowdhury, Smart parking systems: comprehensive review based on various aspects, Heliyon 7 (5) (2021) e07050. doi:https://doi.org/10.1016/j.heliyon.2021.e07050.

[19] A. Khanna, R. Anand, IoT based smart parking system, in: 2016 International Conference on Internet of Things and Applications (IOTA), 2016, pp. 266–270.

[20] L. Mainetti, L. Patrono, M. L. Stefanizzi, R. Vergallo, A Smart Parking System based on IoT protocols and emerging enabling technologies, in: 2015 IEEE 2nd World Forum on Internet of Things (WF-IoT), 2015, pp. 764–769.

[21] C. Yuan, L. Fei, C. Jianxin, J. Wei, A smart parking system using WiFi and wireless sensor network, in: 2016 IEEE Int. Conf. on Consumer Electronics-Taiwan (ICCE-TW), 2016, pp. 1–2.

[22] C. Trigona, B. Andò, V. Sinatra, C. Vacirca, E. Rossino, L. Palermo, S. Kurukunda, S. Baglio, Implementation and characterization of a smart parking system based on 3-axis magnetic sensors, in: 2016 IEEE Int. Instrumentation and Measurement Technology Conference Proceedings, 2016, pp. 1–6.

[23] Z. Zhang, X. Li, H. Yuan, F. Yu, A Street Parking System Using Wireless Sensor Networks, International Journal of Distributed Sensor Networks 9 (6) (2013) 107975.

[24] R. Tandon, P. K. Gupta, Optimizing Smart Parking System by Using Fog Computing, in: M. Singh, P. Gupta, V. Tyagi, J. Flusser, T. Ören, R. Kashyap (Eds.), Advances in Computing and Data Sciences, Springer Singapore, Singapore, 2019, pp. 724–737.

[25] M. Mohandes, N. Tasadduq, S. Aliyu, M. Deriche, A preference based smart parking system: KFUPM case study, in: 2015 31st International Review of Progress in Applied Computational Electromagnetics (ACES), 2015, pp. 1–2.

[26] J. Rico, J. Sancho, B. Cendon, M. Camus, Parking Easier by Using Context Information of a Smart City: Enabling Fast Search and Management of Parking Resources, in: 2013 27th International Conference on Advanced Information Networking and Applications Workshops, 2013, pp. 1380–1385. doi:10.1109/WAINA.2013.150.

[27] J. Shi, L. Jin, J. Li, Z. Fang, A smart parking system based on NB-IoT and third-party payment platform, in: 2017 17th International Symposium on Communications and Information Technologies (ISCIT), 2017, pp. 1–5. doi:10.1109/ISCIT.2017.8261235.

[28] A. Mackey, P. Spachos, K. N. Plataniotis, Smart parking system based on bluetooth low energy beacons with particle filtering, IEEE Systems Journal 14 (3) (2020) 3371–3382. doi:10.1109/JSYST.2020.2968883.

[29] Y. Agarwal, P. Ratnani, U. Shah, P. Jain, IoT based Smart Parking System, in: 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), 2021, pp. 464–470. doi:10.1109/ICICCS51141.2021.9432196.

[30] A. R. Masoud Humaid Al Mamari, H. Al Mamari, S. I. Ali Kazmi, J. Pandey, S. A. Hinai, IoT based Smart Parking and Traffic Management System for Middle East College, in: 2019 4th MEC International Conference on Big Data and Smart City (ICBDSC), 2019, pp. 1–6. doi:10.1109/ICBDSC.2019.8645589.

[31] A. M. S. Maharjan, A. Elchouemi, Smart Parking Utilizing IoT Embedding Fog Computing Based on Smart Parking Architecture, in:

2020 5th International Conference on Innovative Technologies in Intelligent Systems and Industrial Applications (CITISIA), 2020, pp. 1–9. doi:10.1109/CITISIA50690.2020.9371848.

[32] N. Farooqi, S. Alshehri, S. Nollily, L. Najmi, G. Alqurashi, A. Alrashedi, UParking: Developing a Smart Parking Management System Using the Internet of Things, in: 2019 Sixth HCT Information Technology Trends (ITT), 2019, pp. 214–218.

[33] Y. Geng, C. G. Cassandras, New "Smart Parking" System Based on Resource Allocation and Reservations, IEEE Transactions on Intelligent Transportation Systems 14 (3) (2013) 1129–1139. doi:10.1109/TITS.2013.2252428.

[34] G. R. Leone, D. Moroni, G. Pieri, M. Petracca, O. Salvetti, A. Azzarà, F. Marino, An Intelligent Cooperative Visual Sensor Network for Urban Mobility, Sensors 17 (11) (2017). doi:10.3390/s17112588.

[35] G. Amato, F. Carrara, F. Falchi, C. Gennaro, C. Meghini, C. Vairo, Deep learning for decentralized parking lot occupancy detection, Expert Systems with Applications 72 (2017) 327–334.

[36] F. M. Awan, Y. Saleem, R. Minerva, N. Crespi, A comparative analysis of machine/deep learning models for parking space availability prediction, Sensors 20 (1) (2020). doi:10.3390/s20010322.
URL `https://www.mdpi.com/1424-8220/20/1/322`

[37] W. Shao, Y. Zhang, B. Guo, K. Qin, J. Chan, F. D. Salim, Parking availability prediction with long short term memory model, in: GPC, 2018.