Politecnico di Torino
Vodafone Academy

Smart cities project
Smart parking

Sofia Caterini
Alice Maritato
Lina Medina
Davide Auricchio
Giacomo Cerfeda
Gabriele Degola
Nicolas Vega

July 2020

1 Introduction

Data traffic continuous growth due to novel applications that include ultra-high-definition videos, augmented reality (AR), and tactile Internet has brought in new connectivity requirements such as higher bit rates, lower latency, greater reliability, higher capacity, robust coverage, and power efficiency. The fifth generation of mobile wireless networks (5G) has emerged as an answer to these incoming needs.
The promising characteristics of 5G have also inspired the creation of plenty of applications that years ago were limited by the capabilities of the previous mobile generations. The goal of this report is to present the design of a smart parking solution that allows the car drivers to know which are the zones with more probability of empty parking lots and receive instructions on how to reach them in real-time exploiting the 5G technology.
The remaining of this report is organized as follows. Section 2 presents the problem. Section 3 introduces a general view of the smart parking solution proposed in this work. Section 4 reviews the existing detection systems. The analysis of machine learning techniques and the development of the algorithm for the detection of empty parking lots are presented in Section 5. In Section 6 the architecture strategy is analysed. The user experience, the mobile application, and the design of the app are discussed in Section 7, instead in Section 8 we applied the queuing theory to our smart parking system to evaluate the probability of parking in a specific parking lot. Section 9 describes the most common and suited navigation systems that can be integrated into our application. Finally, in the last sections, a general view of the Business model that can be adopted for our smart parking system is analyzed.

2 Problem statement

Finding an empty parking lot could be a nightmare for drivers, especially in peak hours or crowded places such as the industrial or commercial zones. Usually, the search for unoccupied parking lots consumes time and money and makes drivers get more stressed. Moreover, it increases traffic congestion in the cities. The question then arises, "How is it possible to improve the parking search, reducing the time this task takes and therefore decreasing the traffic in the cities?"
The parking search problem requires innovative technological solutions to be addressed. This project introduces a proposal of an application that takes advantage of the features of the fifth generation of mobile

wireless networks to provide information to the users in real-time about the nearest empty parking lots.

3 General proposal

In figure 1, our proposal is summarized. It was decided to include two different systems. The main system is based on high definition cameras that record the parking and wirelessly send the high definition video to a 5G device that transmits it in real-time to the Mobile Edge Computing (MEC). There, the video is processed with machine learning to detect empty lots in the parking. This system (camera-based) is expected to be used in most of the parking, especially those that are big and contain several lots like the one in the upper image of the diagram.

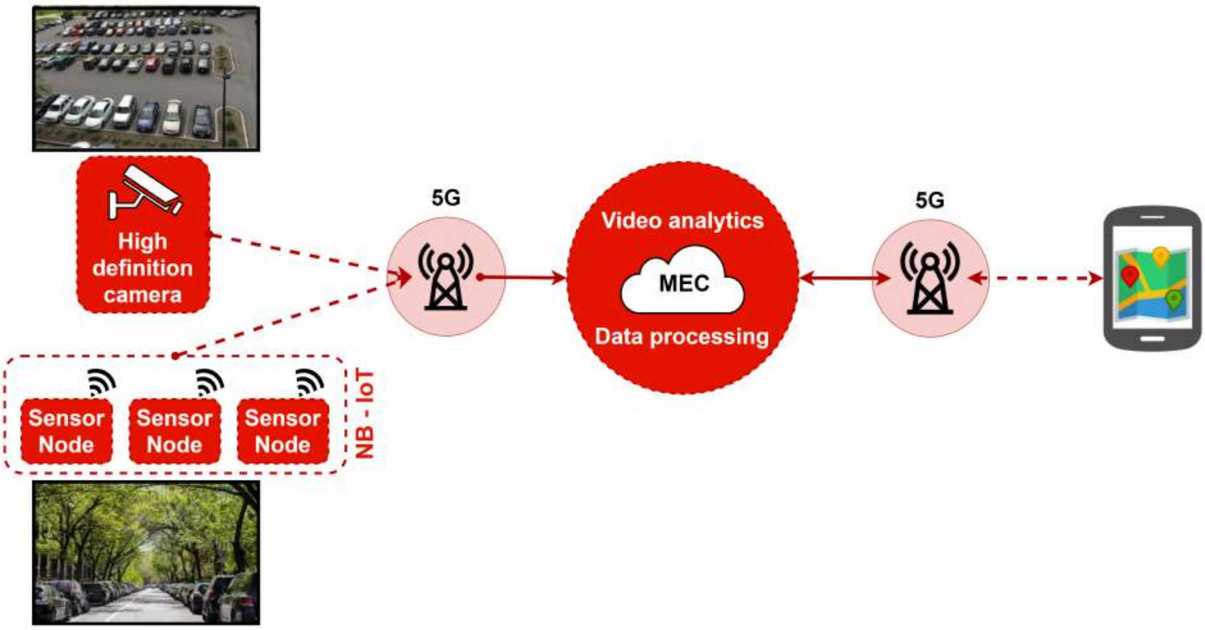


Figure 1: Diagram of the proposed solution.

There are a few cases where it is not possible to achieve camera visibility, particularly in those places where the trees or buildings obstruct the camera like in the image in the button of the diagram. For these cases, we propose a Narrowband IoT sensor network composed of nodes that detect the presence or absence of cars and send the data to the cloud.
From the side of the user, he would use a web or mobile application. In it, he sends his position in real-time to the server in the MEC. There, an application in the server selects the nearest parking zones where the probability of finding an empty lot is the highest. Then, the user can visualize the parking location and receive indications of how to reach it. The driver would also send additional information depending on the navigation system he uses. For instance, in the case of augmented reality, he would send videos of the road in real-time to receive the instructions.

4 Detection systems

4.1 Sensors

In smart parking applications different types of parking sensors can be used. Sensors can be active or passive: the former need of an external power source to accomplish their task, the latter work without an external power supply. [13]
Sensors are evaluated based on the following features:

- **Intrusiveness**: an intrusive sensor is embedded in the pavement, thus its installation and maintenance requires modification of the road or the floor. In the contrast, a non-intrusive sensor is typically mounted on a structure or it is on the side of the road; [12]

- **Weather sensitivity**: the weather conditions may affect the accuracy and the effectiveness of a sensor;
- **Accuracy**: the sensor accuracy is the ability to reliably detect cars;
- **Cost**: cost influences the feasibility and the possibility to use the smart parking application in a large scale, thus the objective is to find a cost-effective sensor, i.e. a sensor which has a low cost and an accuracy which is as good as possible;
- **Maximum sensing range**: range in which the sensor senses the presence or the absence of the vehicle; [8]
- **Size of target**: it is referred to the size of the object that the sensor can sense; [8]

Some examples of active sensors are:

- **Active infrared sensors**: they emit infrared radiation and determine whether a parking spot is available or not based on the amount of radiation reflected. They are sensitive to the weather conditions and they have limited cost and accuracy;
- **Ultrasonic sensors**: they emit short, high-frequency sound pulses at regular intervals which propagate through the air at the speed of sound. When these waves strike an object, they are reflected back as echo signal. The distance to the target is computed based on the time-span between emitting the signal and receiving the echo. They are sensitive to weather conditions, they have low cost and a good accuracy.

Passive sensors are:

- **Passive infrared sensors**: they detect the energy that is emitted or reflected from vehicles or road surface. Similarly to active infrared sensors, this type of sensor is affected by weather conditions and has limited cost and accuracy;
- **Light Dependent Resistor sensors or LDR**: they are also known as photo resistors or photocell. They are resistors whose resistance varies according the amount of the incident light on their surface. When the resistor is exposed to light, the resistance changes;
- **Inductive Loop Detector sensor or ILD**: they consist in several loops connected through cables to the detector unit. In the presence of a car, the inductive field in the loops changes. It is subject to the stress of the traffic and it has a low accuracy despite having a high cost. [12]
- **Piezoelectric sensors**: a car going over the sensor produces a pressure. This pressure generates a voltage difference which is proportional to the weight of the vehicle passing over the sensor. In this way, this sensor detects the presence or the absence of the car; [13]
- **Pneumatic road tube sensors**: they send a burst of air pressure along a tube when a vehicle passes over the tube. The pressure pulse closes an air switch and produces an electrical signal. They have a high cost and high accuracy;
- **Magnetometer sensors**: they sense the changes in the earth magnetic field which is caused by metallic objects, such as vehicles, passing over these sensors. The cause for such distortion is that the magnetic field can easily flow through ferrous metals in contrast to air. These sensors are reliable and resistant even in adverse weather condition. They have low cost and high accuracy. It is an intrusive sensor, since the installation and maintenance requires cutting into the pavement;
- **Microwave radar sensors**: they transmit at frequencies between 1 and 50 GHz through an antenna, which detects the vehicles based on the reflected frequency. These sensors can also work in harsh weather conditions. They have a good accuracy and a high cost;

- **Radio Frequency Identification sensors or RFID**: they consist in three element - transceiver, transponder and antenna. When triggered by an electromagnetic interrogation pulse from a RFID reader device, the tag transmits digital data back to the reader. It can be placed inside the vehicle to identify if it is placed in the parking lot. It is not the most reliable method to detect the availability of a parking spot. These sensors have low cost and a limited accuracy;
- **Acoustic sensors**: they sense the changes of acoustic energy produced by the transit of a vehicle and through a signal-processing algorithm detect the presence or the absence of a vehicle. These sensors have a low cost and low accuracy. [12]

The following table compares the most common types of parking sensor with respect to their most relevant operational requirements. [13] [26] [22] [19].

Sensor type	**Intrusive**	**Weather sensitive**	**Accuracy**	**Cost**	**Maximum sensing range**	**Size of target**
Active infrared	Y	Y	**	€€	200 m	≥ 5 mm
Ultrasonic	N	Y	***	€	4 m	All sizes
Passive infrared	N	Y	**	€€	200 m	≥ 5 mm
LDR	N	Y	**	€	2 m	All sizes
Inductive Loop	Y	Y	****	€€	Depends on the velocity target	All sizes
Piezoelectric	Y	Y	****	€€€	Depends on the weight target	All sizes
Pneumatic road tube	Y	N	****	€€€€	Depends on the weight target	Short vehicles
Magnetometer	Y	N	****	€	Depends on the size target	All sizes
Microwave radar	N	N	***	€€€	40 m	Large vehicles
RFID	N	N	**	€	Depends on the frequency band	All sizes
Acoustic	N	Y	*	€	10 m	All sizes

Since the purpose is to design a smart parking application which is off-street and outdoor and which is far from the traffic, the sensor chosen is the magnetometer: as said before, it has a low cost and it has a good accuracy, it has no problems under harsh weather condition and despite it is intrusive, since the parking lot is far from the traffic, this aspect is not a problem.

4.1.1 Communication Protocol

About the connection of these sensors, in order to reduce the computational cost we decided to add a more simple wireless communication, we think to use one typical IoT protocol.
The category of protocols that we have analyzed, as they are best suited to our type of network architecture, are LPWAN. A low-power wide-area network is a type of wireless telecommunication wide area network designed to allow long-range communications at a low bit rate among things (connected objects).
The main attributes are:

- **Long range**: The operating range of LPWAN technology varies from a few kilometers in urban areas to over 10 km in rural settings. It can also enable effective data communication in previously infeasible indoor and underground location;

- **Low power**: Optimized for power consumption, LPWAN transceivers can run on small, inexpensive batteries for up to 20 years;
- **Low cost**: LPWAN's simplified, lightweight protocols reduce complexity in hardware design and lower device costs. Its long range combined with a star topology reduce expensive infrastructure requirements, and the use of license-free or licensed bands reduce network costs.

There are a number of competing standards and vendors in the LPWAN space, the most prominent of which include:

- **LoRa**: *Good bandwidth, long-range, for private networks, good security, great interference resistance.*

  First, let's clarify that LoRa and LoRaWAN are not the same. LoRa is a proprietary modulation format (designed by Semtech) that transforms the actual physical signal into a form suitable for transmission, while LoRaWAN is an open network protocol that ensures secure two-sided communication.

  LoRaWAN provides reliable bi-directional M2M interaction within a range up to 15 km in the suburban setting, and up to 5 km in densely built cities at the rate of up to 50 kbps. To accommodate the fact that requirements of IoT applications differ significantly, LoRaWAN supports three classes of end-devices depending on the volume and frequency of data exchange. All three classes can coexist in the network, and devices can switch their class dynamically, which favors flexibility and scalability of deployed solutions.

  To guarantee message delivery, LoRaWAN uses acknowledgments that are sent via the downlink (from the server to the end-device). Although generally useful, this procedure has a downside – it reduces the overall capacity of the channel.

  Since a lot of IoT technology solutions deal with sensitive data, security is a major concern in the field. LoRaWAN provides encryption on network, application, and device layers to make the protocol well-suited for protected data transfer.

- **Sigfox**: *Lowest bandwidth, decent range, lowest power use, good security, expanding worldwide.*

  Sigfox was one of the pioneers who realized the potential of ultra-narrow bandwidth (UNB) for IoT communication. The company is nurturing an ambitious plan to become a global IoT operator.

  Sigfox ensures low-bandwidth uplink transmission at up to 300 bps and very limited downlink at only 12 bytes per day. Basically, Sigfox can be used effectively only to gather sensor data, but can't send any commands back to the sensors. Low bandwidth contributes greatly to reduced power consumption – an end-device can operate up to 10 years on a single AA battery. The signal can travel the distance up to 50 km in a rural areas and up to 10 km in a concrete jungle of the large cities

- **Cellular IoT**: Cellular IoT is a way of connecting physical things (like sensors) to the internet by having them piggyback on the same mobile networks as smartphones. Its infrastructural simplicity combined with the dawn of 5G positions cellular IoT as a strong player in the connectivity space. The two most important standards are:

  – **NB-IoT**: *Great indoors and outdoors, low power use, infrastructure serviced by cellular operators, expanding worldwide.*
    NB-IoT (Narrow Band IoT) is an open technology standard developed by 3GPP (3rd Generation Partnership Project) for the Internet of Things applications. It uses a band inside the LTE spectrum and therefore relies on cellular network operators in terms of infrastructure. The fact that NB-IoT uses LTE networks also suggests that it features best-in-class LTE-level security.

    NB-IoT signal travels up to 20 km in rural areas, up to 5 km in the urban setting, and is particularly good at penetrating inside buildings and underground. The standard ensures up to 250 kbps of both uplink and downlink communication and supports up to 100,000 devices per cell (base station).

  – **LTE-M**: *Superior data rates, decent range, low power use, best security.*
    LTE-M (Long Term Evolution for Machines), as well as NB-IoT, is a child of 3GPP, so those two

will be competing against each other for cellular operators' attention. The good news about LTE-M is that it is completely compatible with existing LTE base stations, therefore the infrastructure deployment becomes virtually investment-free.
One of the distinctive functions in LTE-M protocol is a Power Saving Mode, which made it possible to extend battery life up to 10-20 years. Hardware updates are possible over the air but can reduce battery life.
LTE-M guarantees bi-directional data transfer over the distance of up to 5 km at the rate up to an impressive 1 Mbps.

The following table summarize the main features.[4]

	LoRa	**Sigfox**	**NB-IoT**	**LTE-M**
Cellular	No	No	Yes	Yes
Spectrum	Unlicensed	Unlicensed	Licensed	Licensed
Range, km	Urban: 2-5 Rural: 15	Urban: 3-10 Rural: 30-50	Urban: 1-5 Rural: 10-15	Urban: 2-5
Speed, uplink /downlink	50Kbitps/ 50Kbitps	50Kbitps/ -	250Kbitps/ 250Kbitps	1Mbitps/ 1Mbitps
Power consumption	***	*	*	***
Security	**	**	***	***
Availability of devices	**	***	**	*
Prize	**	*	**	***
Areas of application	Precision farming, manufacturing automation, pipeline monitoring	Capacity planning, demand forecasting	Electing metering, static sensors, manufacturing automation	Tracking objects, wearables, energy management, city infrastructure

In conclusion, NB IoT is perfectly suitable for our purpose. Enables a wide range of new IoT devices and services, significantly improves the power consumption of user devices, system capacity and spectrum efficiency, especially in deep coverage. Is easily integrate with 5G mobile networks. And above all, is being touted as the potentially less expensive option, because it eliminates the need for a gateway so sensor data is sent directly to the primary server.

4.2 Camera

Camera is a detection system which can be used both in outdoor and indoor parking and which is especially suited for large parking areas: it allows a cost-effective implementation because with a single camera you can analyze more than one parking spot.
The video, which is acquired by the camera, is transmitted to a computer and then it is digitized and analyzed through image processing techniques.
Furthermore, several parking lots are already equipped with surveillance cameras which can be used to implement a detection system.
Weather conditions and the presence of obstacles may affect the performance and the effectiveness of this detection system.

4.3 Drones

Another way to detect the availability of a parking spot is using drones. Drones or UAVs (Unmanned Aerial Vehicles) are aircrafts designed to operate autonomously or to be piloted remotely without a pilot on board.[1]

They can be equipped with several sensors and they allow to have all needed sensors in one device, solving problems related with the installation and maintenance of ground sensors.
Furthermore, drones provide a larger detection range with respect to a single camera, as well as high mobility, high precision, flexibility.
However, in low light or bad weather conditions their performance could get worse or they may not to be able to fly at all. Also, in the case of coexistence of more than one team of drones in the same area, a collision avoidance system is needed.
Other aspects to take into account before considering to use drones in a smart parking system are their technical limitations: a professional flight drone has a flight time of 30-40 min using two batteries and a full charge requires 90 min. Moreover, its operating range is 7-8 km.
Another factor to consider in the design of smart parking which involves the use of drones is the analysis of legislation in force. Up to the 1st January 2021 in Italy the ENAC (Ente Nazionale per l'Aviazione Civile) legislation is in force, while after this date the European Commission Regulation About UAVs (EU 2018/1139) will come in force.
According with ENAC legislation, three different types of operations exist which the operator can perform:

- **VLOS (Visual Line of Sight)**: the drone can only fly in the visual range of the operator;
- **EVLOS (Extended Line of Sight)**: the visual range of the operator is extended using alternative devices which are recognized as valid by the legislator;
- **BVLOS (Beyond Line of Sight)**: operation in which the operator does not use visual contact to guide the drone. [3]

According with EC legislation, only two types of operations exist:

- **VLOS (Visual Line of Sight)**: the remote pilot maintain continuous visual contact with the drone, allowing the remote pilot to control the flight path of the drone taking into account other aircraft, people and obstacles avoiding collisions;
- **BVLOS (Beyond Line of Sight)**: operation which is not conducted in VLOS. [2]

The operations required by a practical smart parking are included in BVLOS category in both legislation. While it is theoretically possible implement smart parking application without operating in BVLOS, this would be unpractical as it would require the constant presence of a skilled operator for each drone.
Considering the type of operation in which drones can be involved, ENAC legislation defines two types of operation:

- **non critical**: operations in VLOS in which drone does not fly over gatherings of people, urban areas and critical infrastructure;
- **critical**: operations not included in the previous definition. [3]

EC legislation distinguishes three types of operations according to the related risk:

- **open**: operations which have the lowest risks and which do not require prior authorisation;
- **specific**: operations which have medium risk. These operations may be carried out in VLOS or BVLOS. Risk assessment is required;
- **certified**: operations which have the highest risk. They involve the transportation of people or the carriage of dangerous goods and they may constitute a high risk risk for third parties is case of accident. They are subject to rules on certification of the operator and the licensing of remote pilots. [2]

The use of drones for smart parking application is an operation in critical scenario (since the type of operation is a BVLOS) according to ENAC legislation, thus it requires:

- compliance statement of the operator with respect to the ENAC legislation;

- an adequate level of reliability compatible with the type of operation to do;
- flight termination device;
- appropriate risk analysis to demonstrate ad adequate level of safety.

Moreover, flying over crowds, such as areas where there are shows and event, is forbidden. [3]
According to EC legislation, the smart parking application falls in the category of specific operation, since it requires BVLOS operation. It requires:

- the risk assessment and the robustness of the mitigating measures to keep operation safe during the flight;
- operational authorisation from the competent authority.[2]

In conclusion, using drones for smart parking applications requires authorizations, risk analysis and the analysis of the technical limitation which UAVs have, thus nowadays drones should be used only in specific cases of smart parking solutions, such as temporary parking areas which are designed for festival.

5 Computer vision

High definition cameras are particularly effective to detect available parking slots in wide areas with few obstacles, where they can cover a large number of parking slots with a relative low cost. In this situation Computer Vision algorithms can be used to extract information from a real time video and label each slot as available or occupied, solving a binary classification problem.

5.1 Related works

In the literature, the parking detection problem is mainly approached in two ways. Tătulea *et al.* [24] and Almeida *et al.* [6] manually segment the parking slots from each frame of a recorded video and use extracted features, such as gradients and local patterns, to train a support-vector machine classifier. In addition, [6] introduces the PKLot dataset, a very large collection of parking lots images collected by three different cameras. Other works use deep learning solutions to progressively extract features from the images at different abstraction levels. Cazamias and Marek [11] use a simple Convolutional Neural Network to automatically learn correlations between adjacent pixels and classify the segmented lots of PKLot, obtaining very high accuracy scores. Similarly, Amato *et al.* [7] exploit a simplified AlexNet, which runs directly on surveillance cameras, and introduce CNRPark, a parking spots dataset containing frames collected by nine different cameras in different weather conditions and with widely occluded spaces.

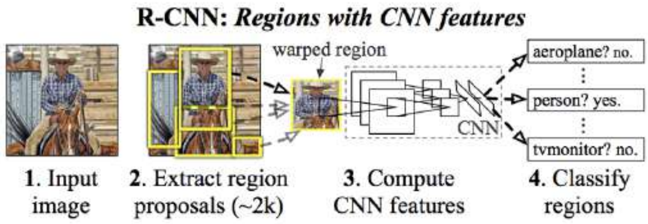
(a) PKLot (b) CNRPark (c) CNRPark

Figure 2: Images from the PKLot and CNRPark datasets

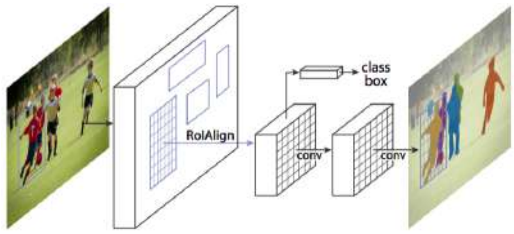
5.2 Methodology

The manual segmentation of parking slots is a complex and costly operation and becomes infeasible for systems that deal with a large number of parking areas. For this reason, we adopted a solution based on a state-of-the-art algorithm for object detection. Similarly to Cai *et al.* [9], we used a Mask R-CNN to automatically detect the actual parking slots and learn their position in the images. Once they are stored somewhere in the Mobile Edge Computing, the same network can analyse a real time video and detect if a vehicle covers a spot, classifying it as available or occupied.

Mask R-CNN: Mask R-CNNs have been proposed in 2017 by He *et al.* [17] and represent an evolution of region-based Convolutional Neural Networks, which are widely used for object detection. In basic R-CNN [15], a manageable number of possible regions of interest (RoI) are extracted from an image and a CNN is independently evaluated on each RoI to extract features which are feed into a SVM to classify the presence of an object in that region and the bounding box location. Fast R-CNN [14] and Faster R-CNN [21] improved the region selection algorithm, analysing only interesting RoIs and reducing exponentially the required time, making the network suitable for real-time applications. Mask R-CNN adds a branch to Faster R-CNN to return an object mask, together with the class label and a bounding box. This operation is very simple and adds a very little overhead. For these complex networks, the role of the main CNN or *backbone* is very important, as it must extract accurate and useful features. As in the original paper we used a ResNet with 101 convolutional layers, a CNN with very high performances.



(a) Description of R-CNN



(b) Architecture of Mask R-CNN

Figure 3

Finally, the Mask R-CNN model has to be trained, and it does require a very large number of images of the objects we want to detect. Luckily, we aim at the detection of vehicles, especially cars and trucks, which are very common objects included in several public dataset. For example, the Microsoft COCO dataset [18] contains over 200000 images annotated with object mask, with more than 12000 car images. We developed our project in Python, using mainly the open source libraries OpenCV [20] and Mask_RCNN [16], which is implemented in Keras and includes a model pretrained on COCO. An example of its application is reported in figure 4. At this point, we can require the bounding boxes for vehicles only, obtained with a reasonable confidence.

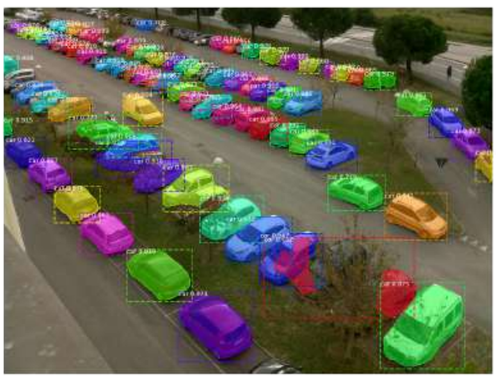


Figure 4: Visualization of Mask R-CNN's output on CNRPark

Our proposal: We tested two methods for parking slots detection on the CNRPark dataset, using the presented architecture. At first, we imagined to have available a "reference image" for each parking, in which

Algorithm 1: Parking detection algorithm

```
Input: List of frames
Output: List of identified parking slots
parkings ⟵ ∅
i ⟵ 0
Create new FIFO queue q
foreach frame do
    /* save position of vehicles in the current frame */
    boxes ⟵ getCarBoxes(frame)
    push boxes in q
    if i ≥ n then
        /* if at least n frames are processed, extract the oldest car boxes and
           compute the overlapping with the already detected parking slots */
        oldBoxes ⟵ pop from q
        computeOverlapping(oldBoxes, parkings)
        foreach boxes in q do
            /* consider next frames and compute overlapping with previous potential
               parking slots */
            potentialParkings ⟵ ∅
            computeOverlapping(oldBoxes, boxes)
            foreach box in oldBoxes do
                get maximum overlapping with other boxes and with stored parkings
                if maxOverlappingBoxes > t_c and maxOverlappingParkings < t_s then
                    /* if a car box satisfies the threshold it is stored as potential
                       parking slot */
                    potentialParkings ⟵ potentialParkings + box
                end
            end
            oldBoxes ⟵ potentialParkings
        end
        /* car boxes detected in n subsequent frames are stored as actual parking
           slots */
        parkings ⟵ parkings + potentialParkings
    end
    i ← i + 1
end
```

every parking space is occupied by a vehicle. In this way the object detection algorithm can be executed on one frame only, but it must be carefully chosen in order to cover all slots. However, the result may be flawed because of missing detections of the R-CNN or due to the presence of non-parked vehicles in the image, as well as incorrectly parked vehicles.

Therefore we developed a more complex algorithm for parking spot detection, whose pseudocode is reported in algorithm 1. Working with CNRPark, we tested the algorithm choosing one of the nine cameras and considering the frames acquired on a single day to identify the parking slot, testing the result on the other frames for that camera. The algorithm iterates on the training frames, ordered by acquisition time, and a slot is extracted if a vehicle is detected in a certain position for at least n consecutive frames. For each frame, the position of the vehicles is returned by the R-CNN and the Intersection over Union (IoU) measure is computed for each extracted vehicle, with respect to the positions extracted from the next $n-1$ frames and to the already identified parkings. IoU is computed dividing the amount of overlapping pixels between two bounding boxes by the total number of pixels covered by both boxes, as represented in figure 5.

Finally, an extracted bounding box is saved as an actual parking slot if the maximum IoU between the box

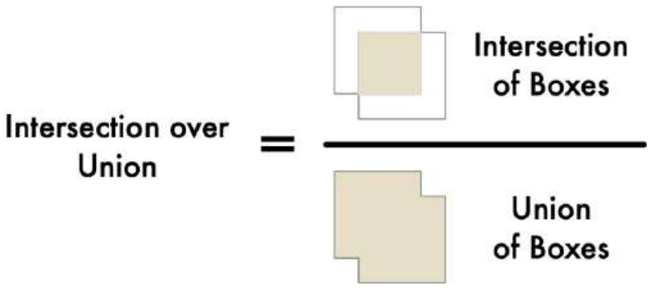
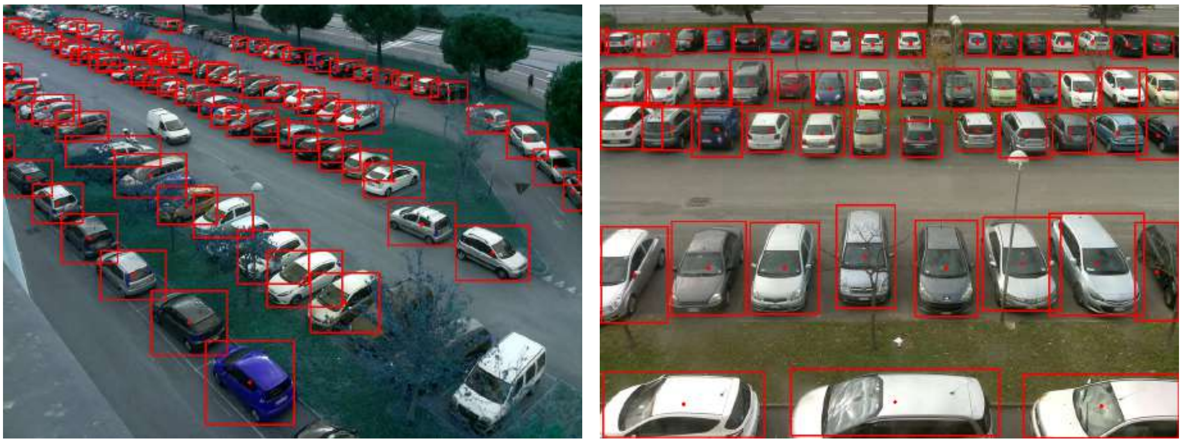


Figure 5: Intersection over Union measure

and the boxes of the next $n-1$ frames is above a threshold t_c and at the same time the maximum IoU between the box and the already stored parkings is below a second threshold t_s. The results of the algorithm on two different cameras are visualized in figure 6.



(a) Camera 1

(b) Camera 8

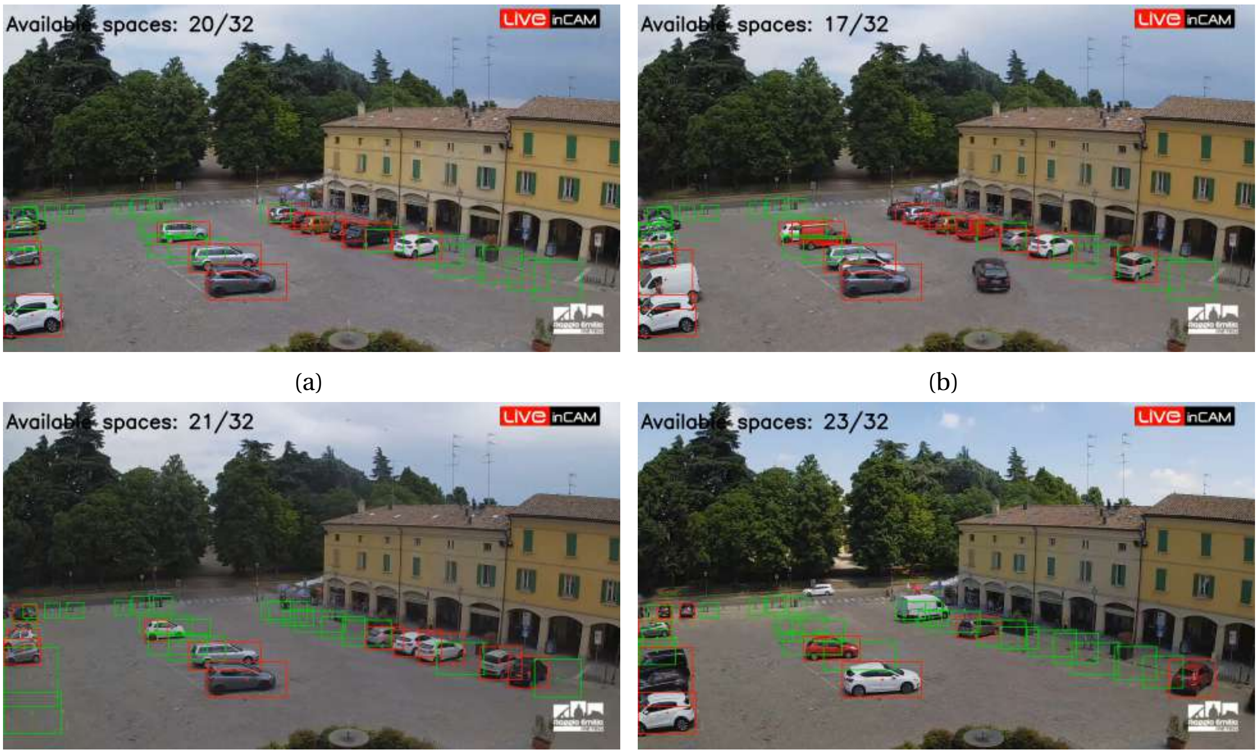
Figure 6: Extracted parking slots from two different cameras

At this point, one test frame is processed at the time and a parking slot is signalled as occupied if the maximum IoU overlapping with the boxes extracted from the new frame is above t_c. In the following images, free slots are highlighted in green and the occupied slots in red.
From the previously reported images, it is evident that our algorithm is able to detect most parking slots, in particular in figure 6b, where the camera is well positioned and there are few occlusions. In figure 6a this is more difficult, in particular for the farthest slots which are occluded by other vehicles. For these reasons, it is important to accurately choose the parameters of the algorithm to adapt it to different parking spots. For example, for camera 1 it is necessary to use a higher t_s with respect to camera 8, while the number of consecutive frames n can be low for parkings in which the average parking time is short, for example a supermarket, while it can be higher in other situations, for example in airports, to extract parking slots with the best possible accuracy and in a reasonable time. In addition, a possible problem arises after processing the frames of camera 1: if a vehicle is parked in an area in which parking is illegal it is obviously stored as a parking slot. That issue must be addressed in postprocessing or setting a sort of mask before running the algorithm.

At this point, it is straightforward to apply our parking detection algorithm on a video stream. As we don't have a direct interface to a live video stream, we worked on a publicly available live webcam provided by the municipality of Cavriago, Reggio Emilia [25], which is particularly suitable for our purpose. We recorded some videos of variable length from the live streaming and used them for training and testing, extracting one frame every 10 seconds and processing them as described before. Once the parking slots are located they can be printed on a video, signalling if they are available or occupied. Some result frames are reported in figure 8.

(a) Camera 8 (b) Camera 8

Figure 7: Classification results



(a) (b)

(c) (d)

Figure 8: Frames of a result video

Most of the parking slots are correctly identified and correctly labelled, although it is not perfect especially for the parkings on the left of the video, which are often occluded by vehicles parked in other slots. Derived problems can be addressed using a probabilistic approach, while if a highly accurate detection is required it is necessary to install cameras in the best location to cover the entire parking spot without occlusions.
The presented algorithm can be easily applied to a real-time video in a real scenario. When a new camera is installed it is used in a first phase to identify the parking slots, and then it can give real-time indications about the parking status. The length of the first phase should be evaluated depending on the nature of each parking.

6 Architecture strategy

The first part of the work about the architecture is finding a solution to implement the smart parking starting from the main components from the user devices to the sensors and so on:

- **Camera**
- **Sensors**
- **Elaboration unit**
- **User terminal**

About the sensors and the cameras there are some alternatives that we analyzed:

- **De-centralized architecture**: the de-centralized model has an high reliability, because we can use the devices to do a part of the elaboration. There is a problem, it is the huge cost to realize devices, because we need to realize ad-hoc devices. It's more difficult to realize, devices need to coordinate.
- **Centralized architecture**: the centralized architecture is the selected one, it's the best solution in terms of cost/benefits. Here we selected to have a central 'server', but for our purposes there is an edge computing. In that way we have an high reliability. Devices, like cameras, can be implemented using a 5g router, no need to realize embedded devices.
- **Asyncronous architecture**: the asyncronous way can be used only to have an alternative in a non 5G scenario. Here we lost a lot of the advantages of the edge computing. For example, image processing algorithm are improved with the experience, with a periodic manner of retrieving data we lost a lot of data to improve our algorithm.

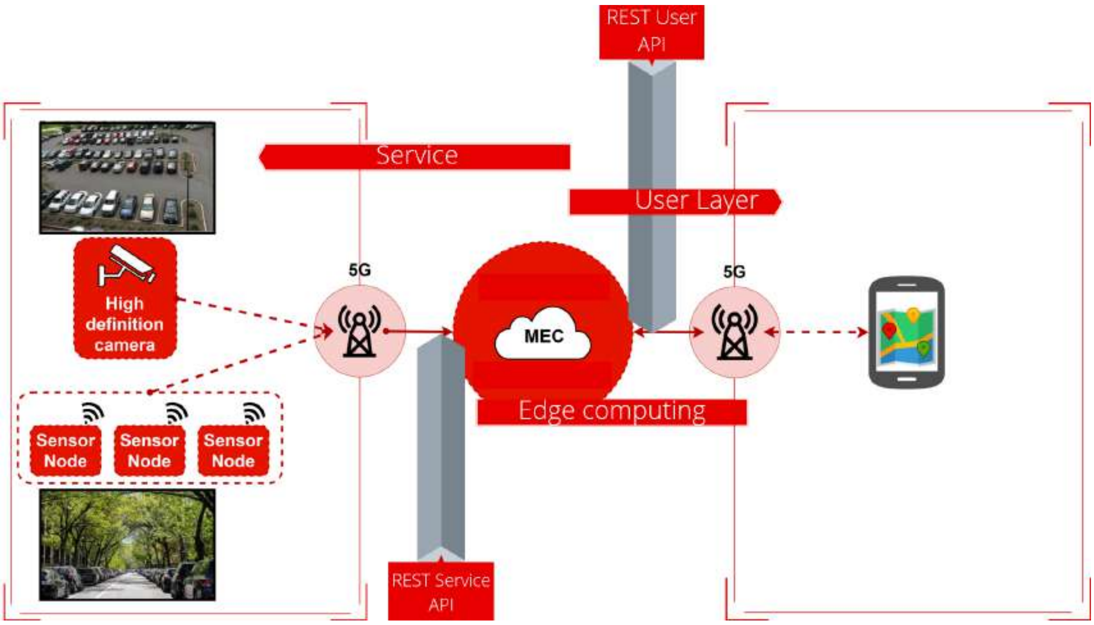


Figure 9: Architecture schema

With our choice, the centralized architecture we have:

- **Show of 5G points of strength**: 5G use on connection between user and our middleware to provide AR services with a low latency and 5G with ultra wideband for connection between an huge amount of cameras with an huge amount of data to stream.
- **Data analyzed in real time on the cloud**

- **No embedded devices to create**
- **Cloud point of failure can be minimized up to 99.9 percent**
- **We can convert old devices like cameras only using a 5G router**

Why 5G between Camera/Sensors and middleware? We need to send to the middleware an huge amount of data (HD video stream) with a little power consumption. 5G Ultra-wideband is the radio technology that can use a very low energy level for short-range, high-bandwidth communications over a large portion of the radio spectrum. For example, to stream a 30 Megabytes video (about 2 seconds of a 4K video) we have to wait about 8 seconds in 4G, in the 5G case we have to wait less than a second. Imagine multiplied for n video streams it becomes an huge advantage the use of the 5G technology.

The part about the user-middleware interaction can be summarized in a web application using AR on Web (for example using ARCore from Google) and there is the possibility to sell to developers the API used for the realization of the project, like in the Google Maps for developers case.

Why AR in this project? The 48 percent of all consumers say they will use AR in the next five years across different media types and with that implementation we can give to the user an immersive experience. "The internet connection must also be excellent, with high bandwidth and very low latency, or else the AR experience will suffer!" say Ericsson experts. This is exactly what 5G is designed to do. Network slicing, distributed cloud, and edge computing technologies will together guarantee stable performance and ultra-low latency.

7 User experience

An important consideration has to be take into account when we study the functionality of the system in relationship with the drivers. As reported in figure 1, the smart parking system interfaces with the users through a **mobile/tablet application**. In this section we will look at all the features of the application and the solutions proposed for the several complications that rises when we have to deal with users. A quote of Jakob Nielsen impeccably describe the importance of a good design and usability of a web/software application: *"On the Web, usability is a necessary condition for survival. If a website is difficult to use, people leave. If the homepage fails to clearly state what a company offers and what users can do on the site, people leave. If users get lost on a website, they leave."*

What we have to take into account are those parameters that do not depends from sensors or from a computing system: we are dealing with different characteristics that are quite uncontrollable like

- **The probability that during a movement a park denoted to a user as free can be occupied by another driver;**
- **The probability that during a movement the user could decide to change its destination, to stop or just take a break before reaching the park;**
- **The probability that an already noted as free park lot to a far user (journey > 20 minutes) can be useful and available to a user which is closer to the same park lot;**
- **The coverage of the app in a city among drivers and the amount of them that really use it.**

To help us on achieving a reliable system that is able to give a better experience to users, we need cameras and magnetic sensors which provide us an enormous amount of data about all the park areas of a city. How this data are processed is discussed in the section 8. For the moment suppose that we know all about park lots and all the characteristics of a specific park lot. Now let's see why and how we use it.

7.1 Notification of available park lots

A driver that would like to find a park lot with the smart parking system application have to open the app, share the GPS position, set the destination and the maximum radius in which he would like to park. Figure 10 shows the first approach to the mobile application.
At this point the application displays all the available park lots through a position mark, like in Figure 11. Different colours of the position marks highlights the different probability of finding free that park lot during the route to destination. We cannot give for granted that a specific park lot will remain available for the user. In this way the user is informed about the possibility to find free that park lot.

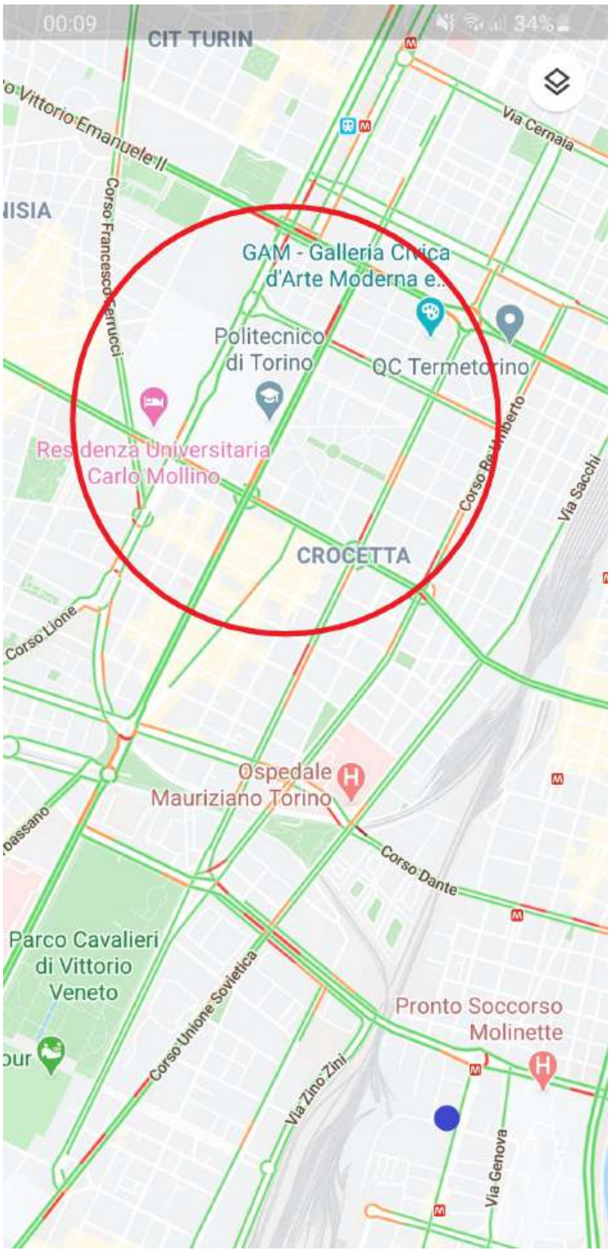


Figure 10: General view of the app.

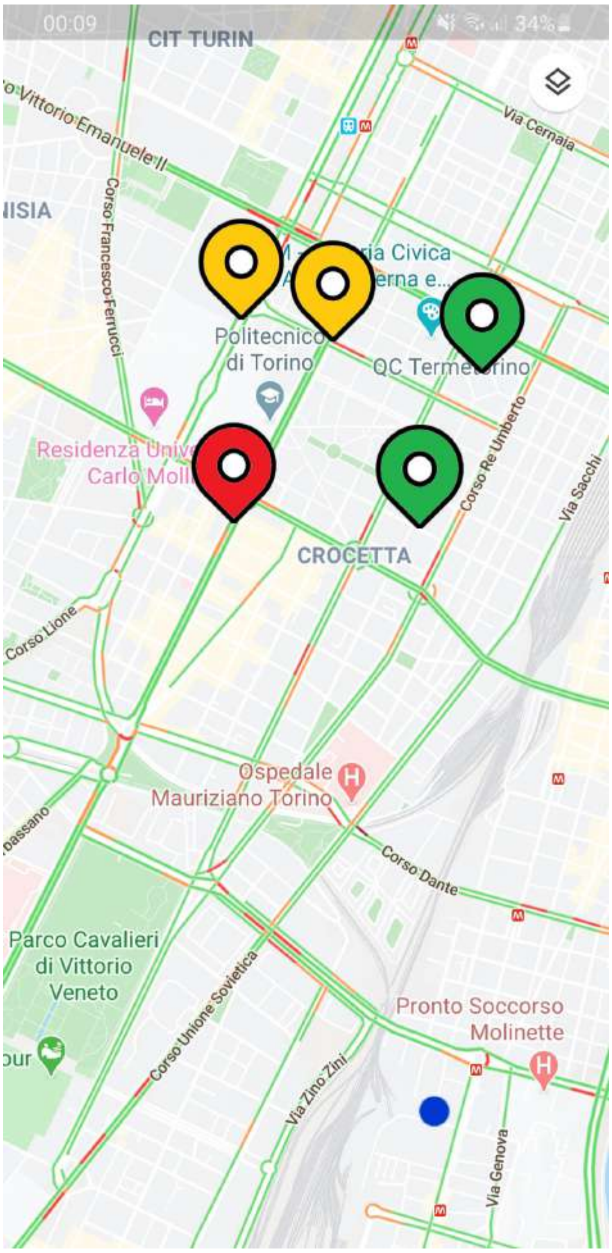


Figure 11: Coloured position mark indicate the available park lots.

The three colours are green, yellow and red and the associated probability with respect to them is like:

- **green position mark:** from 66% to 100%;
- **yellow position mark:** from 33% to 66%;
- **red position mark:** from 0% to 33%.

The probability that assign a colour to the position mark can change according to different factors.

7.2 Better route direction

The user, at this point, is able to see all the available park lots and can decide to reach one of them. If the park lot chosen by the user have medium-low probability to be free, the system is able to suggests a route directions which cover different park lot, instead of going directly to one only. In this way, the possibility to find a free park lot increase. Figure 13 shows a possible suggestions that the system give to user instead of the one only park route directions, like in Figure 12.

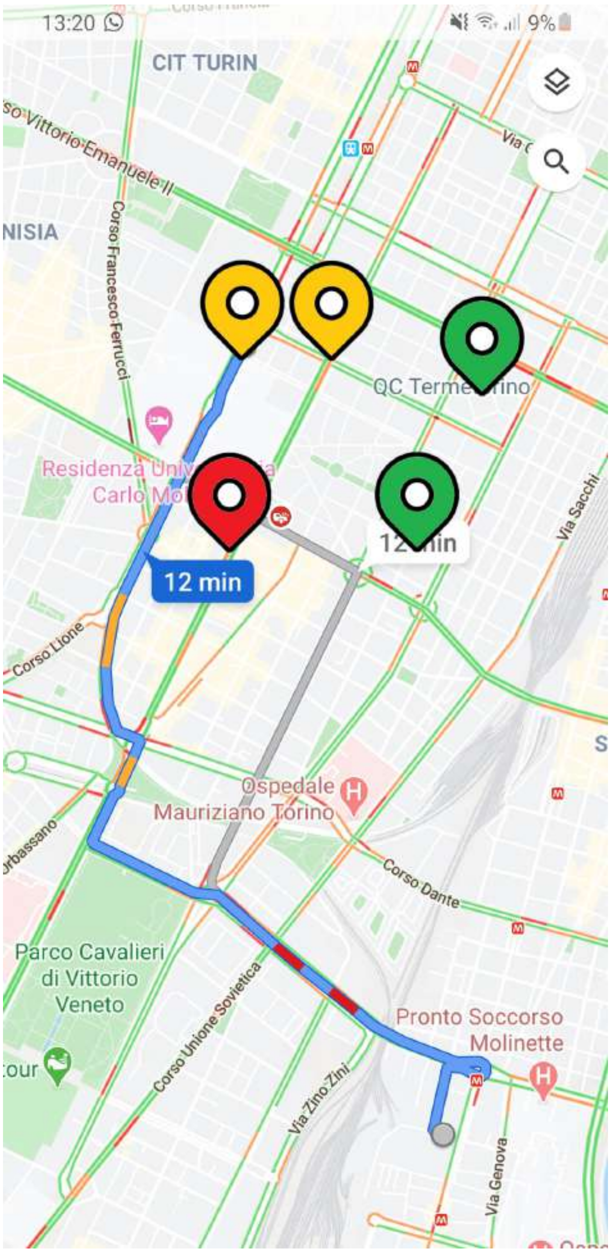


Figure 12: Route direction to one-only park lot.

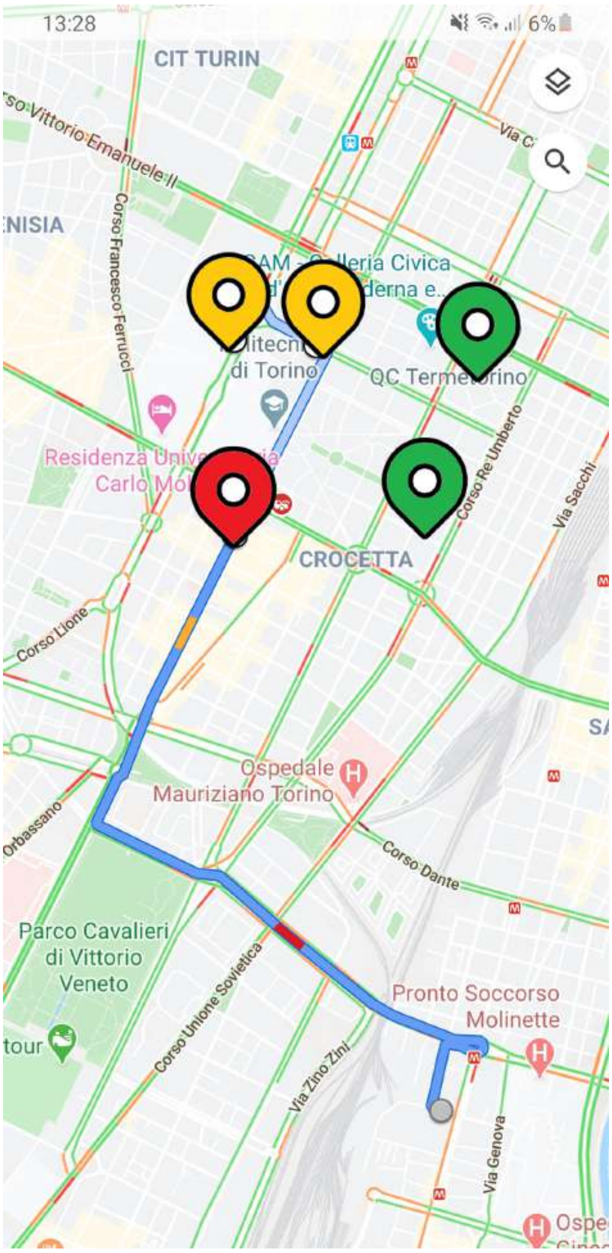


Figure 13: Route direction to multiple park lot.

7.3 Closer user served approach

A generic system or web service is mainly based on ***"First arrived first served"*** approach. A user that open the app before with respect to another user has the priority on receiving first the services provided by the application. This implies that we need a **waiting list**: users will be served according to the time that they enter into the application.

This approach is completely useless for our smart parking system: assume that the blue user, shown in Figure 14, distant from Politecnico (our destination) about 20 minutes. He will open the app, choose the park lot available and he start to reach the destination. In this amount of time we can have that another user, for example the black one, which is distant to the same park lot of about 5 minutes, is not able to find a park for reaching the same destination. Using the FAFS approach we have this situation: the closer user is searching a park and since from the app that park lot is set as full he will continue to search, consuming time, increasing traffic jam and increasing the pollution. Now looking at the farther user, may happen that until he arrives at destination, another driver can have occupied that parking lot.

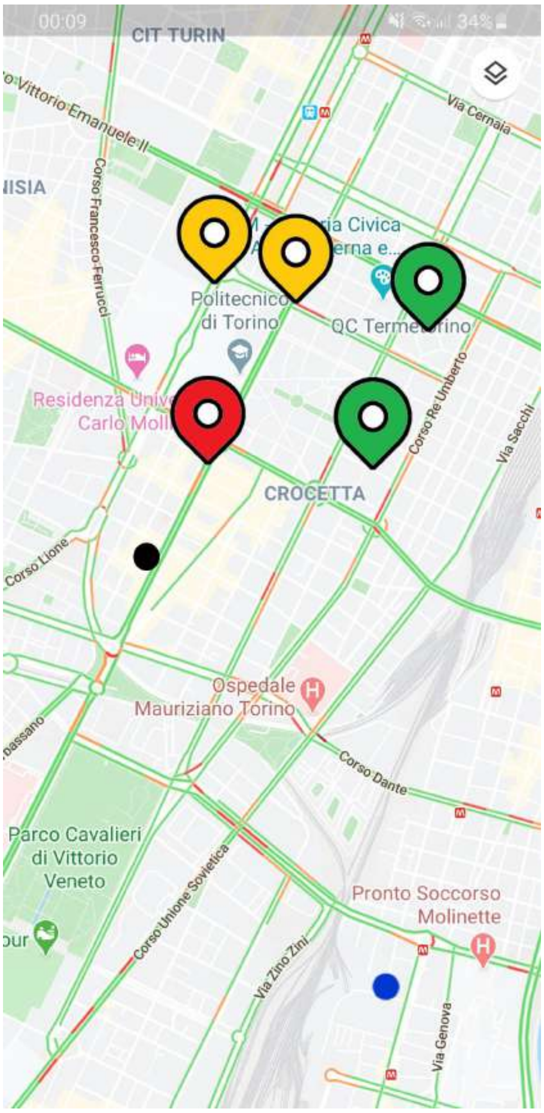


Figure 14: Closer user and farther user.

Instead, using a ***"Closer user served"*** approach we give the opportunity to both the user to increase the possibility of parking: the closer user can see the same park lot of the farther user, so he can reach the park lot and since is very close to the park the probability of parking are quite high with respect to the farther user. In the same time we may have that a park lot near the destination, that initially was full, could loose some car. At that moment, the application will notify the farther user, which is reaching the destination, that a park lot, near to the destination is now available.

7.4 Spread of the application

Larger is the number of the drivers that use the application better will become the experience and the control of the system. Nowadays this is very difficult because we cannot force drivers to use the application and also because not all the population have a smartphone.

In the future, when smart car will become popular, the usability of the application will improve, also because it can be installed directly on the car system. The tracking of the car and the experience of the user will be enhances since GPS can be placed inside the car.

8 Evaluation of the parking probability

The probability that we assign to a park lot in order to indicate the effective possibility of parking is a crucial part of the system: if this estimation is not properly done, the whole structure go down.

The model from which we start to have an idea of how to evaluate it is the ***Queue model***, based on queuing theory. Usually this model is used for describing call centers or the service time of a router with the buffer that represent a waiting list. The most suited queue for our project is the ***M/M/C/C Queue*** in which we have no waiting list and the C represents the available park lots. The drivers arrive according to a Poisson process with rate λ and the service times (or parking time) have an exponential distribution with rate μ. λ and μ are tuned according to the previous collected data from camera and sensors. They depends on different parameter like:

- **Distance from the current position**
- **Hour**
- **Day**
- **Middle of the week or week-end**
- **Season**
- **Weather condition**
- **Events near the destination**
- **Traffic jam**
- **and so on**

Once that the data are collected, we can use several machine learning algorithm trough which λ and μ can be obtained, even in those cases in which the parameters was obtained in different situation.
In Figure 15 we can see that this model perfectly suited for our smart parking system.

A park can be modeled as a Markov chain with C states like in Figure 16. Each state represents the number of customers (drivers) in a park lot. Every time that a car arrives we move to the next state; vice versa if a car leave the park. A drivers is lost when the park lot has already C car.

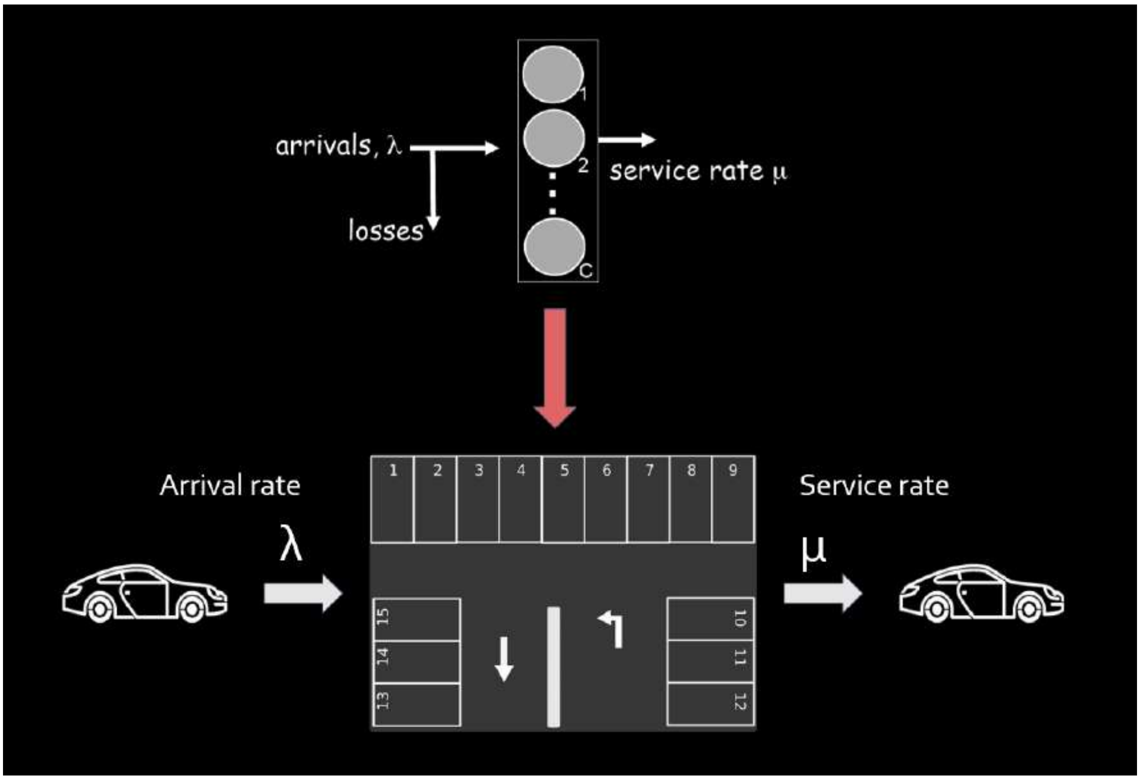


Figure 15: M/M/C/C model of a park lot.

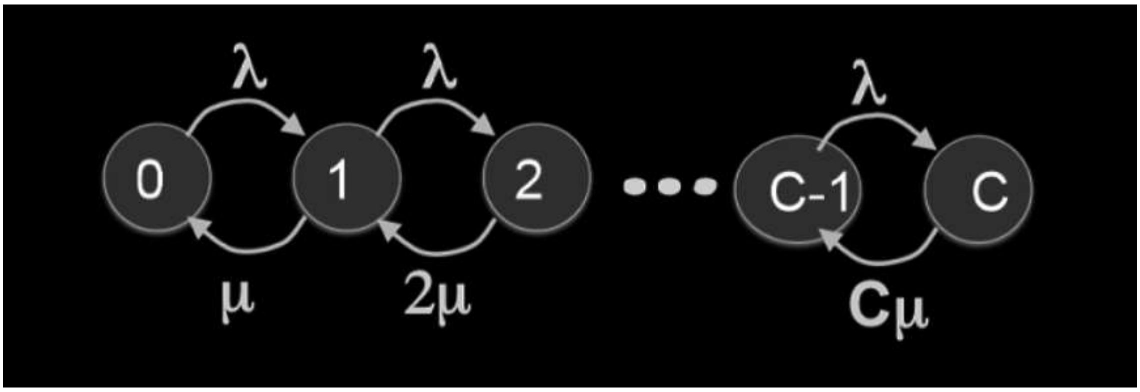


Figure 16: Markov chain of a park lot.

With the help of **Markov chain** and all the information that we have about a park lot, its arrival time, its service time and the actual occupied park lot we can evaluate this probability. Recall that our aim is to evaluate the probability that a user will find an empty park lot.

Assume that we have:

- **A park with C park lots**;
- **k park lots are occupied**;
- **User search a park at time t0**;
- **User distant t1 minutes from that park**.

We denote ***Pk,C as the probability that a Markov chain will be in state C at time t=t0+t1 in the future, given that at time t0 the park is in state k***.

A ***Q-matrix*** is used in order to be able to calculate the transition probabilities from time t0 to time t=t0+t1. The transition probability take into account all the possible combination that we can have from state k (k park lot are occupied) to state C (park lot is full).

In other words ***the probability (the one that assign the colour to the position mark) is evaluated as the probability that in the t1 minutes required by the user to reach that park areas, the k occupied park lot will become equal to C.***

If this probability goes from 66% to 100% a red color is assign to the park lot, meaning that potentially the park lot could become full before user arrive at that park. Instead if it goes from 33% to 66% a yellow position mark is choose and, in the end, if it goes from 0% to 33% a green mark is assign, representing a park lot which give the possibility to easily park.

The just discussed section is better explained in these papers. [10] [23]

9 Navigation system

Till now we have analysed all the mechanics related to the detection of a free parking lot and how to indicate it to a user. At this point the user would like to reach this park and here comes the navigation system which guide him to the park lot.

In our project we have studied and analysed three different systems that we consider the most important ones: **the Computer Graphic navigation systems, the Hud Based and Augmented Reality one.**

The first one is based on giving the route directions through a 2D map. An example of this approach is Google Maps, Figure 17: the user simply chooses the park lot near its destination and then with the GPS position and through the route directions is able to reach the destination.

Figure 17: Computer graphics navigation system.

The **Hud Based** system use, instead, the windscreen of a car like a projector, in which we can display different parameters like the velocity, the gear, and most importantly for our case, the directions that we have to take. We use smartphone or tablets like a projector which are lay below the windscreen, in order to show a part of a cross with the arrows that indicates the directions. Often the windscreen is replaced with a sort of a screen directly attached to the smartphone, like in Figure 18.

Figure 18: Hud based navigation system.

The **Augmented Reality** navigation system is the newest technology in this field: the images of the street that we are passing through are captured in real time with the smartphone camera, which is directly pointing to the windscreen and so on the street. This real time video/images are interpolated with the route directions: the smartphone, which display is pointing towards the driver, shows a videos of the street with the arrow for the directions integrated on the same real time video. Figure 19.

Figure 19: Augmented reality navigation system.

The goggle maps approach is obviously the technology which is well known nowadays since we are using it from different time and we learn how to use. The disadvantages of this navigation system is that when we reach a cross, maybe a cross with different exits, we can have problem on understanding which is the one that we have to choose.

The Hud Based system, instead, is very good in the cases in which we are near a cross. Since the information are on the windscreen we are subjected to less distraction, something that Computer Graphics navigation system and in the Augmented Reality cannot ensure since we have to take our eyes off the windscreen. We mainly have 3 disadvantages for this systems:

- **images on the windscreen are subjected to weather condition**: for example when we have a sun which is shining the info becoming less visible, vice versa when we have clouds;
- **info which are displayed cover a part of the windscreen**: this give less visibility for the driver;
- **difficulty on understanding the route directions when we are distant from a cross**: in general we do not have the complete view of the whole path.

We think that the Augmented Reality is the best solution for a navigation system from the point of view of the user. In fact it is easier to understand the directions when we are near a cross and also during the route navigation, something that neither the Computer Navigation system nor the Hud Based are good.
For this system we need good resolution camera in order to make high quality videos and so make easier the representation of a street. Nowadays smartphones are able to do that. The only disadvantages that we have found is that it can distract the user from driving.
The Hud Based system and Augmented Reality can be integrated with the technology which are already implemented on a car. Nowadays, in fact, different car companies provides in series with a car a projector, which is able to display some information like velocity etc, or cameras which are installed in front of a car with the aim of helping the driver in low visibility condition. Our application could collaborate with the car and with these technologies for improving the navigation experience of the user. [5]

10 Stakeholders

Define the way that the service will be offered requires the identification of the main stakeholders involve in the value chain:

- **The Italian Data Protection authority**: the new regulation related with the protection of personal data (GDPR) plays a big role in all kind of processors of data. Since the solution is providing a service through the collection of data from sensors and cameras, the SmartParking solution should be carefully aligned with the practices for design and development which allow to minimize any risk of data breach that can cause potential damage to the data subject.
- **The customers**: who from a general view can be public entities such as universities, municipalities and hospitals and also private ones like parking operators, shopping center, retail and supermarkets. However, the solution tend to focus on those who are more likely to provide a worst driver experience due to high and dynamic traffic in crowded areas. Therefore, the customer are public entities, in particularly, those units in charge of the public traffic and parking administration such the GTT (Gruppo Torinese Trasporti) in the case of the city of Turin.
- **The drivers**: who are those that are looking for parking, having as potential result the reduce of anxiety and stress, which can be critical factor in the user decision at the moment to select the customer over their competitors.

11 Business model

To break down and explain every single aspect of the business model, it is taken the Canvas model approach in order to align stakeholders and encourage the discussion of further potential trade-offs of the business model proposed.

- **Key partners**: Sensor suppliers, camera suppliers, automotive companies, especially those that involve self-driving cars initiatives, public bodies/units in charge of the public traffic and parking administration.
- **Key Resources**:

  Physical: Cameras, sensors, servers, data storage and telecommunication infrastructure.

  Intellectual: Softwares, architecture, designs and trademarks.

Human: Telecommunication engineers, software developers, UX/UI designers, project managers, data scientists and data engineers.

- **Key Activities**: Maintenance, software validation, algorithm validation, UX/UI validation, achieve high accuracy, MVP.
- **Value Proposition**: Increase the visitor or user experience, in specific, provide a better support to find a spot available for drivers in crowded places, by using a friendly and smart assistant.
- **Customer Relationship**: The business model presented is very similar as the one used for Internet Service Providers, who take care about the maintenance and installation, and they sell the access to the Internet using their infrastructure and charging a monthly fee for it.
- **Market segments**: We consider only public or private bodies in charge of the public traffic and parking administration. Then, it's more easy to estimate the size of market by analyzing the population in major cities of Italy. Assuming that cities with a population over 100.000 can suffer from traffic congestion and parking availability issues, it implies 45 big projects with one or more monthly subscriptions (only in Italy).
- **Channels**: Word of mouth between public and private bodies, interregional conferences, public contests.

Finally, two versions of the service can be offered:
The first one is a end-to-end solution, which is the most appropriate one having in mind that our customer is not specialized. In the same way, the service includes many of the core businesses of Vodafone such as 5G, IoT, Cloud services and, of course, Software development which is include in many product that Vodafone offers. In that way, we can add additional features that are easy to implement like payments and pricing.
Moreover, this service will allow to gather huge amounts of data, which can be processed for decision-making for Vodafone and the customers for example, it can help local authorities to decide whether to expand or reduce parking slots, or for instance, again, setting a real-time pricing.
On the other side, another option is by implementing a SDK, because can attract third parties to foster innovation around the service and avoid lock-in. In the same way, gives the possibility to provide customised solution for customers. Finally, it opens a gate for further future integration with key partners for example for self-driving cars.

References

[1] Official journal of the european union. L 212/1, 2018.

[2] Official journal of the european union. L 152/48, 2019.

[3] Regolamento mezzi aerei a pilotaggio remoto. ENAC, 2019.

[4] Lpwan comparison, 2020.

[5] K. Akaho, T. Nakagawa, Y. Yamaguchi, K. Kawau, H. Kato, and S. Nishida. Route guidance by a car navigation system based on augmented reality. *Mitsubishi Electric Corporation, Japan Nara Institute of Science and Technology, Japan Osaka University, Japan*, vol.2, 2012.

[6] P. Almeida, L. Soares de Oliveira, A. Jr, E. Jr, and A. Koerich. Pklot - a robust dataset for parking lot classification. *Expert Systems with Applications*, 42, 02 2015.

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

[8] Banner. Vehicle detection sensors, 2020. Last accessed 6 July 2020.

[9] B. Y. Cai, R. Alvarez, M. Sit, F. Duarte, and C. Ratti. Deep learning-based video system for accurate and real-time parking measurement. *IEEE Internet of Things Journal*, 6(5):7693–7701, Oct 2019.

[10] M. Caliskan, A. Barthels, B. Scheuermann, and M. Mauve. Predicting parking lot occupancy in vehicular ad hoc networks. *Volkswagen AG, Wolfsburg, Germany -murat.caliskan, andreas.barthels- at volkswagen.de*.

[11] J. Cazamias and M. Marek. Parking space classification using convolutional neural networks. *Stanford University*, 2018.

[12] M. Chowdhury and A. Sadek. *Fundamentals of Intelligent Transportation Systems Planning*. Artech House ITS library. Artech House, 2003.

[13] A. M. Fadi Al-Turjman. Smart parking in iot-enabled cities: A survey. *Elsevier*, 2019.

[14] R. Girshick. Fast r-cnn, 2015.

[15] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation, 2013.

[16] Github - matterport/mask_rcnn. `https://github.com/matterport/Mask_RCNN`.

[17] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn, 2017.

[18] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár. Microsoft coco: Common objects in context, 2014.

[19] P. Misra, S. Jha, and D. Ostry. Long-range detection in acoustic sensor networks. pages 372–373, 01 2010.

[20] Opencv. `https://opencv.org/`.

[21] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks, 2015.

[22] S. Solutions. Selection or design of sensor targets, 2020. Last accessed 6 July 2020.

[23] H. Tavafoghi, K. Poolla, and P. Varaiya. A queuing approach to parking: Modeling, verification, and prediction. *Department of Mechanical, Electrical and Computer Science Engineering, University of California, Berkeley*, August 2019.

[24] P. Tătulea, F. Călin, R. Brad, L. Brâncovean, and M. Greavu. An image feature-based method for parking lot occupancy. *Future Internet*, 11:169, 08 2019.

[25] Webcam cavriago | reggio emilia meteo. Available at `http://www.reggioemiliameteo.it/webcam-cavriago.php`.

[26] Wikipedia. Radio-frequency identification, 2020. Last accessed 6 July 2020.