System to Monitor Vehicles in Apartments Garage by SmallAutonomous Car using Deep Learning and Beacons

Leonardo León-Vera[1]   Felipe Moreno-Vera[1]
Renato Castro-Cruz[1]   Jose Navío-Torres[1] and Marco Capcha-Mansilla[2]
{lleonv, felipe.moreno.v, racastroc, jnavio}@uni.pe and
cm.marco919@gmail.com

[1] Universidad Nacional de Ingeniera, Lima, Perú
http://www.uni.edu.pe
[2] Universidad Tecnológica del Perú, Lima, Perú
https://www.utp.edu.pe/

Abstract. Currently, there is an increase in the number of Peruvian families living in apartments instead of houses for the lots of advantage; however, in some cases there are troubles such as robberies of goods that are usually left at the parking lots or the entrance of strangers that use the tenants parking lots (this last trouble sometimes is related to kidnappings or robberies in building apartments). Due to these problems, the use of a self-driving mini-car is proposed to implement a monitoring system of license plates in an underground garage inside a building using a deep learning model with the aim of recording the vehicles and identifying their owners if they were tenants or not. In addition, the small robot has its own location system using beacons that allow us to identify the position of the parking lot corresponding to each tenant of the building while the mini-car is on its way. Finally, one of the objectives of this work is to build a low cost mini-robot that would replace expensive cameras or work together in order to keep safe the goods of tenants.

Keywords: Localization · Self-Driving · Low energy · License Plate Recognition · Bluetooth · Low cost minirobot · License plate detection.

1   Introduction

Peruvian families prefer to live in apartments (see Fig. 1), since the common advantages are more comfortable spaces and less effort in cleaning unnecessary rooms; however, the size of the apartments is reduced every year. Therefore, families decide to keep their goods at parking lots next to their vehicles or inside of them, as long as they are in an underground garage, but some of these stuff are stolen. Also, people who do not live in the buildings use the parking lots

that are only for building users, sometimes thieves parked in the building garage before a stealing inside the building.

A smart self-driving mini-car is proposed to supervise the vehicles in the underground garage by identifying their owners using computer vision in their respective parking lots during a schedule that does not bother the tenants and cars. Using autonomous driving techniques and object detection, a computational vision system is proposed in order to achieve the autonomous navigation of the mini-car and the identification mainly of the cars owners with an ALPR (Automatic License Plate Recognition) based on a deep learning model that was developed by the open source organization OpenALPR. Also, the mini-car has its own location system with Beacons that determine its relative position in real time and it can verify if a vehicle would be at its correct parking lot or not.

Vehicles of the future will be strongly dependent on the development of the software algorithms controlling the autonomous navigation and how they can process the massive amount of data generated, so this mini-car wants to be a prototype of that type of vehicle that is able to be autonomous and at the same time translate the license plates that recognize into an important information for security purposes.
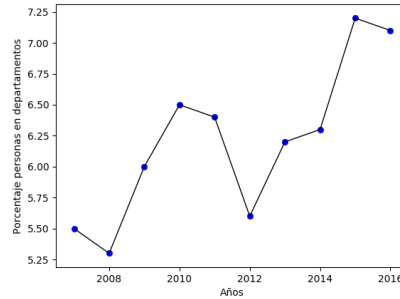

Fig. 1: Growth of people living in apartments. Source: Private households of Peruvians 2007-2016. INEI.

The navigation of the mini car in [14] is based on images recognition using deep learning, developed for more complicated tracks than an underground garage and it uses different sensors for better accuracy in indoor scene.

2   Structure of the System

Implementation of a mini-robot that runs through an underground garage and verify that the tenants cars are in their respective parking lots. It is proposed to solve the problem by detecting objects to identify the vehicle position and its license plate to obtain information from the owner of the car. The holistic

system is made up of three main parts: The computer vision system that allows the self-driving and the license plate recognition; the location system that allows us to know the mini-car location and which parking lots are near at the same time; and the system that describes how the mechanic and electronic mini-car components are integrated.
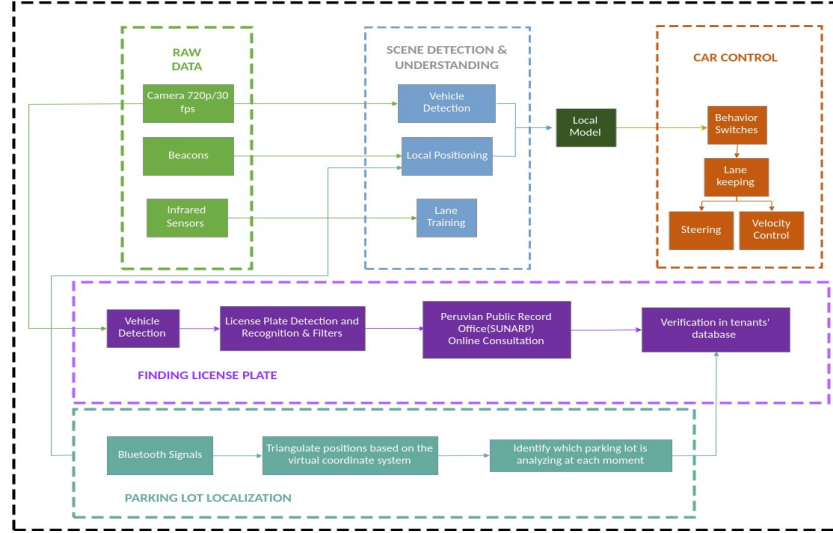

Fig. 2: Structure of the system joining each task.

2.1 Data collected from tenants of the building.

The first step to build a well-designed system that verifies if the vehicle parked on its respective parking lot is having the owner information. This data is collected from the tenants of the building such as the apartment number, name, ID of parking lot, type of vehicle, license plate, objects they stored in the parking lot, etc (See Fig. 3). The owners name and their license plates will be compared with the data obtained by the mini-car license plate recognition and shown to the caretaker so that he can draw his conclusions if a bizarre event happens.

2.2 Autonomous Navigation of the small car

To face this task we use logitech C920 web cam which is cheaper than Lidars and Radars. These are very used for self-driving training since they can calculate the distance between objects with high precision. In contrast, it turns out that web cams emulate the way in which people can see the environment giving a better classification and interpretation of images textures in comparison to the previous

ID Depart.	Name	ID parking lot	Car model	License Plate	Car	Bicycle	...
1101	Pedro	23	Toyota	ACM-123	Si	No	...
1502	Juan	34	Gol	ABC-345	No	Si	...

Fig. 3: Data of tenants.

ones [4]. We are imitating Behavioral Cloning method described in Udacity Self Driving Course because there is only one way where the mini car can go. I mean, It only has to turn left when it is going down and turn right when it is going up. It is one constraint about the garage.

A Convolutional Neural Network based on the NVIDIA model[1] for regression is used to predict the angle of rotation of the wheels of the mini robot only obtaining an image of the path (See Fig. 9).

Data Collection Being a supervised learning, it was necessary to collect a video recording the route that the mini car must follow to obtain an optimal performance. Being a flat floor, we did not get additional problems; however, when the garage walls are very similar, it is needed to look for a correct camera position and cut out the image so that it only focus on taking images that correctly identify curves of straight roads. The input data are the images of three front cameras (images from one camera and 2 images modified) separated from each other by few centimeters to collect more data from the road and speed of the small car each moment; and angles between 0 and 180 as output values for training the model.

Neural Network Architecture A deep sequence model of layers is used in the following manner, with its respective number of filters: Conv24-Conv36-Dropout-Conv48-Dropout-Conv64-Dropout-Conv64-Dropout-FC-FC-FC-FC with non - linear activation function ReLU, which bring good results in computational vision tasks in the Convolutional layers. Regularization methods such as Dropout with a probability of 0.5 are added, due to the overfitting that occurs with the NVIDIA model when iterations are increased considerably.

Neural Network Training and Testing The training of the data was done with the board NVIDIA P4000 with a partition of 20% validation set and 80% training set. The MSE loss function and the ADAM optimizer method are used without the need to manually set the speed of the learning rate.

Testing of the CNN is performed on the Udacity simulator, obtaining a clear outstanding performance when driving autonomously as can be seen in the video [5] and later applied to Jetson TX1 in an underground garage.

2.3   Vehicles Detection

The first attempt to find objects in each parking slot was cars detection. A convolutional neuronal network is in charge of this task using Tiny YOLO model[3] that is the model with the best performance between accuracy and inference time in detection objects in real time such as cars (our priority), bicycles and others objects (see Fig. 10) trained using VOC dataset of 2012. The model also detects void spaces if probability to find any object is lower than the threshold. Once the car is detected, its position is saved in jetson's memory with relative position brought by the beacons.

Neural Network Architecture Tiny YOLO model consists in a convolutional neuronal network with 9 convolutional layers of 16, 32, 64, 128, 256, 512, 1024, 512, 425 filters each one. This model is lightweight in comparison with "YOLO9000: Better, Faster, Stronger" [3] and accors.

Fig. 4: Mini-Robot for testing in the garage

Neural Network Test Test of the neural network was performed on an NVIDIA P4000 GPU server with images taken in the garage and on the NVIDIA Jetson TX1 board which we used in the mini-robot in the garage with a C920 camera obtaining 15 fps which works without any problem with the tiny YOLO model.

2.4   License Plate Recognition

After trying cars detection and getting acceptable results, OpenALPR library has been tried to detect the plate license of each car and later to compare with SUNARP dataset.

The mini robot records videos on its way in the garage so that the license plate recognition could be done.

After obtaining many candidates, we are going to select the license plate that have six digits (Peruvian's license plate rule) and the major often weighted.

After recognizing the characters (letter and digits) of the license plates, the system must access the Peruvian public records office (SUNARP) online

query[10] to collect information related to the owner of the vehicle (see Fig.5). Once important information such as name and license plate is obtained, it will be compared with the database that the mini robot stores from collecting data of the building tenants.

Fig. 5: Query of the owner of license plate in SUNARP online application[10].

2.5   Mini-robot mechanism

Mini-robot is built with base of a Monster Truck 1/10 which includes a motor and 2 servos for the movement of front and rear tires. An Arduino UNO is added that allows communication between the engine and the NVIDIA Jetson TX1 board, as you can see in Fig 4. The Arduino sends instructions to the engine through its GPIOs where indicates speed of the wheels and angle of rotation of the front wheels. Communication of the board TX1 to the Arduino is serial through a USB which allows to send integer values encoded in characters to indicate the rotation of the front wheels and the instructions to go forward, stop and rewind. Through Python program on the TX1 board, instructions are sent to the Arduino and this sends to the motor which angle have to turn.

2.6   Tenant Parking Mapping by Beacons

Summary In this section, we present the way that robot knows where is it in real time. This action can be regulated with Beacons that is useful for indoor location in places where the Internet is not enough to make a connection [7], even allowing tracking in real-time applications [8].
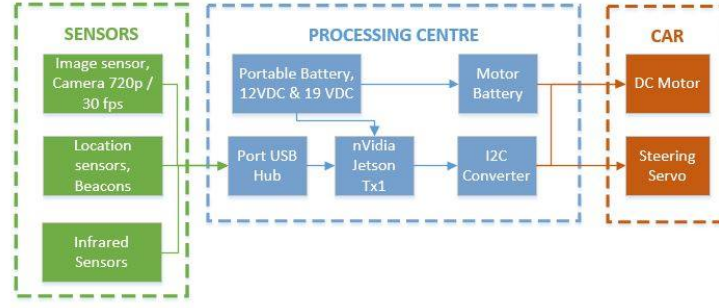
Fig. 6: Hardware communication.

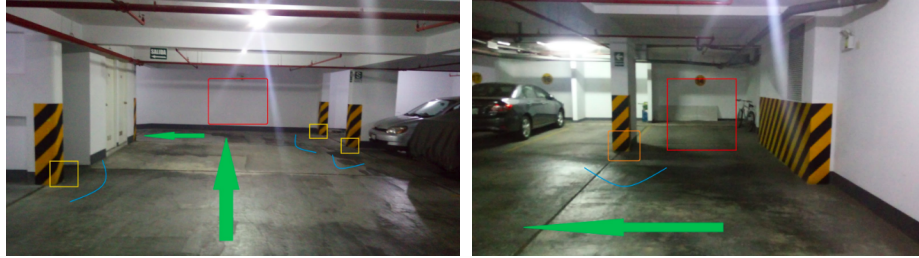(a) Detection of objects, walls and signals of beacons. See the colors: yellow (beacons), red (object detection field), green (direction of route) and celestial (Bluetooth signal).

(b) Positioning of the beacons in the parking lot. See the colors: yellow (beacons), red (object detection field), green (direction of route) and celestial (Bluetooth signal)

Fig. 7: Environment and Beacon's location map

Detection of Beacons In the garage, there is a distance about 8 meters be-
tween walls and 2.45 or 7.35 meters between vehicles, as shown in the Fig. 7(a)
and Fig. 7(b).

As it is seen, Beacons have been distributed efficiently (see Fig. 8) that allows
to calculate where it is located in real time, as well as identifies who owns that
area due to the match between image recognition and the current position, based
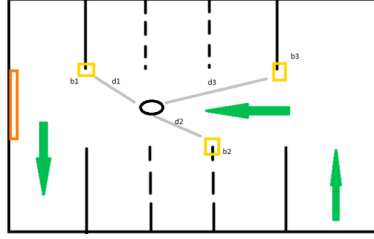on previous work [9].

Fig. 8: Locations of the beacons in the garage and variables to be taken for the
calculation and identification of the beacons.

Determination of the position of the mini-robot This section is about the
equations that define the system to solve, by forming imaginary circles around
each beacon, the respective radii are calculated. Then we proceed to calculate the
relative position with respect to the global system of beacons based on distance
equations forming a system of equations with the following form:

$$E_i : (x - x_i)^2 + (y - y_i)^2 = d_1^2$$
$$\text{for i=1, ..., 3}$$

Where (x, y) is the current position of the mini-robot and the indexes i
corresponding to the beacons and $r_i, ..., r_3$ to the respective vectors.

For this system formed proceeds to solve: Take (x_i, y_i) as coordinates of each
beacon, we deduce $r_i = r_c + d_i$.

Where r_c is the current position of the mini-robot, for all equations the pair
of indices i, j are not equal $(i \neq j)$.

So, The module is taken:

$$\|r_i\|^2 = \|r_c\|^2 + 2(r_c)(r_i) + \|d_i\|^2.$$

Calculating:

$$\|r_i\|^2 - \|r_j\|^2$$

The result is:

$$r_c(d_i - d_j) = \|d_j\|^2 - \|d_i\|^2 + \|r_i\|^2 - \|r_j\|^2 = Y_i$$

By which we would have:

$$x_c(x_i - x_j) + y_c(y_i - y_j) = Y_i$$

Which forms a new linear system that is solved by numerical methods and represented as:

$$Y = AX.$$

Where:

$x = (x_c, y_c)^t$: is the column vector of the mini-robot positions.

A: is the matrix forms by row vectors.

$$A = \begin{bmatrix} x_1 - x_2 \ y_1 - y_2 \\ x_2 - x_3 \ y_2 - y_3 \\ x_3 - x_1 \ y_3 - y_1 \end{bmatrix}$$

Y: is the column vector of substractions between beacons.

3   Results

3.1   Determination of the position of the mini robot

To solving the Linear Equation System, we compared different methods, like iterative and directly. In directly methods we have the problem that the order of the algorithm is almost $O(n^3)$ like Gauss Elimination and derivate should be applied to special matrix (like symmetric, semi positive, positive, dominant diagonal, etc).

Using Methods like Gauss Elimination and iterative methods like Jacobi and Gauss-Seidel (first verify if is a diagonal dominant matrix to apply) iterative methods), we parallel using OpenMP and CUDA. Using Montercalo methods to get an execution average time and approximate a pseudo real solution time.

Method	Serial time	Parallel time (n=2)	Parallel time (n=4)	Parallel time (n=8)	Parallel time CUDA
Gauss-Seidel	0.0729176	0.0670215	0.0362759	0.0152195	0.0000512
Jacobi	0.366089	0.358308	0.370612	0.0151156	0.0000586
Gauss	68.8792	34.4715	17.9784	17.0431	2.321465

Table 1: Execution average time for 40000 times

3.2   Self-driving

After training about 150 epochs we saw that the curve of loss function won't be lower. We got 0.014 of Error in validation and training set using Udacity simulator. Next weeks, we are going to get results of our data obtained.
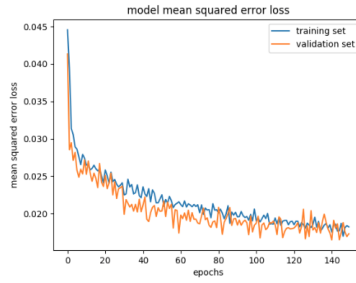
Fig. 9: Graphics of demand and training.

3.3   Results of Cars Detection

Detection of objects is very well identifying cars of general classes (See Fig. 10), however when it is required to identify brand and model of the car there are no results since there is no dataset of images of front or back of the cars with annotations of their marks from which the neural network can learn. The nearest set of images found is Stanford dataset[13], but the majority of images are not taken from front or back view. The model of car recognition was trained with this dataset, but the images used for test were unlabeled. So, this dataset was discarded for the model.
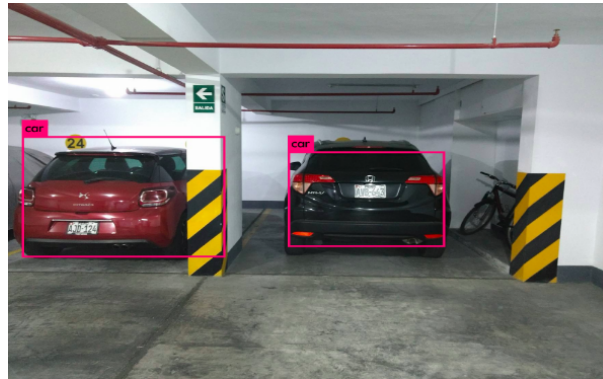

Fig. 10: Detection of cars by mini robot

3.4   Results of Automatic License Plate Recognition

OpenALPR library has a good performance in plate recognition, characters segmentation, and characters recognition [11]. OpenALPR has been trained on its own dataset, however there exists other datasets for this task such as SSIG

dataset [15], a commercial dataset or UFPR-ALPR dataset[2] recently made with more fully annotated images and more vehicles in real-world scenarios for academic purposes.

As it can be seen (See Fig. 12), the actual license plate was recognized with a high total confidence. It turns out that the input is a video file that was recorded by the mini-robot and its outputs in the console (See Fig. 11) are lots of candidates to be considered as the real license plate. After it, three candidates with the most high confidence are chosen to be verified in the tenants database and the SUNARP online consultation.

Fig. 11: License plate recognition of a video recorded by the mini-car
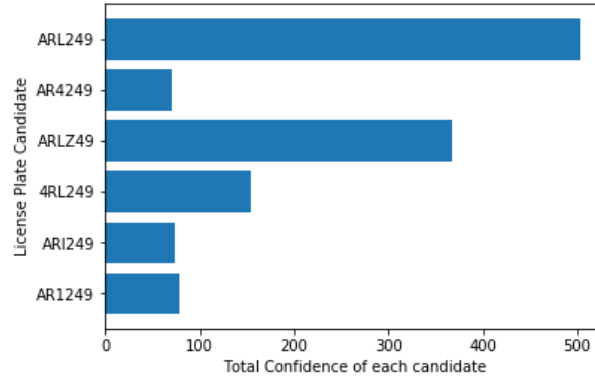

Fig. 12: License Plates vs Total Confidence

4 Conclusions

We see that during the implementation of the robot car, it has to count on many things that had not been foreseen to be able to handle the autonomous car.

In the work procedure, an optimal performance is obtained for the object detection and the automatic license plate recognition using Artificial Intelligence algorithms.

For the determination of positions of the mini robot based on the terrain delimited by columns and walls, it is necessary to specify the locations and the numerical system that solves the system of equations generated in such a way that a minimum error is obtained, which serves to determine and to identify the place where it is when we detect cars and objects of a certain car park, generating the relation object detected - location.

References

1. M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, K. Zieba, End to End Learning for Self-Driving Cars, arXiv preprint arXiv:1604.07316, 2016.
2. R. Laroca, E. Severo, L. A. Zanlorensi, L. S. Oliveira, G. R. Gonalves, W. R. Schwartz, D. Menotti. A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector. CoRR, vol. abs/1802.09567, 2018
3. J. Redmon, A. Farhadi. YOLO9000: Better, Faster, Stronger, CoRR, arXiv preprint arXiv:1612.08242. 2016.
4. D. Santo, 'Autonomous Cars' Pick: Camera, Radar, Lidar?', 2016. [Online]. Available: https://www.eetimes.com/author.asp?section_id=36&doc_id=1330069. [Accessed: 7- Jun- 2018]
5. CarND Self-Driving Nanodegree Udacity. 'Behavioral-Cloning simulation'. [Online]. Available: https://youtu.be/e7D23Kdy72Q. [Accessed: 7- Jun- 2018]
6. CarND Self-Driving Nanodegree Udacity. 'Behavioral-Cloning Project'. [Online] https://github.com/udacity/CarND-Behavioral-Cloning-P3. [Accessed: 18- Sep- 2018]
7. Jinglong Li, Meifeng Guo, Siwei Li. An indoor localization system by fusing smartphone inertial sensors and bluetooth low energy beacons.Frontiers of Sensors Technologies (ICFST), 2017. Shenzhen, China.
8. Ievgen Gorovyi, Alexey Roenko, Alexander Pitertsev, Ievgen Chervonyak, Vitalii Vovk. Real-time system for indoor user localization and navigation using bluetooth beacons. Electrical and Computer Engineering (UKRCON), 2017 IEEE First Ukraine Conference.
9. Ryoya Momose, Tomoyuki Nitta, Masao Yanagisawa, Nozomu Togawa, An accurate indoor positioning algorithm using particle filter based on the proximity of bluetooth beacons. IEEE 6th Global Conference on Consumer Electronics (GCCE), 2017.
10. SUNARP, Servicio Gratuito de Consulta Vehicular con datos relacionados a las caractersticas de los vehculos registrados a nivel nacional. Available: https//:www.sunarp.gob.pe/ConsultaVehicular. Last accessed 7 Jun 2018.
11. C. T. Hsieh, Liang-Chun Chang, K. M. Hung, Hsieh-Chang Huang. A real-time mobile vehicle license plate detection and recognition for vehicle monitoring and management. Joint Conferences on Pervasive Computing (JCPC), 2009.

12. Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV), 2015.

13. Jonathan Krause, Michael Stark, Jia Deng, Li Fei-Fei. 3D Object Representations for Fine-Grained Categorization. 4th IEEE Workshop on 3D Representation and Recognition, at ICCV 2013 (3dRR-13), 2013.

14. Chengmin Zhou, Fei Li, Wen Cao. Architecture design and implementation of image based autonomous car: THUNDER-1. Multimedia Tools and Applications, 2018. DOI: 10.1007/s11042-018-5816-9.

15. G. R. Goncalves, S. P. G. da Silva, D. Menotti, W. R. Schwartz. Benchmark for license plate character segmentation. Journal of Electronic Imaging, vol. 25, no. 5, pp. 053 034053 034. 2016.