Monitoring and surveillance in car parks using computer vision and deep learning

NeurIPS 2018
LatinX in AI

Felipe A. Moreno
Leonardo León et al.
National University of Engineering

22 de enero de 2019

Case of study
León et al

- Increasing of peopleś preferences to live in apartments instead of houses.
- That means people with cars uses the car parks or rent a space.
- Some cases of disappear or stolen things between neighbors.

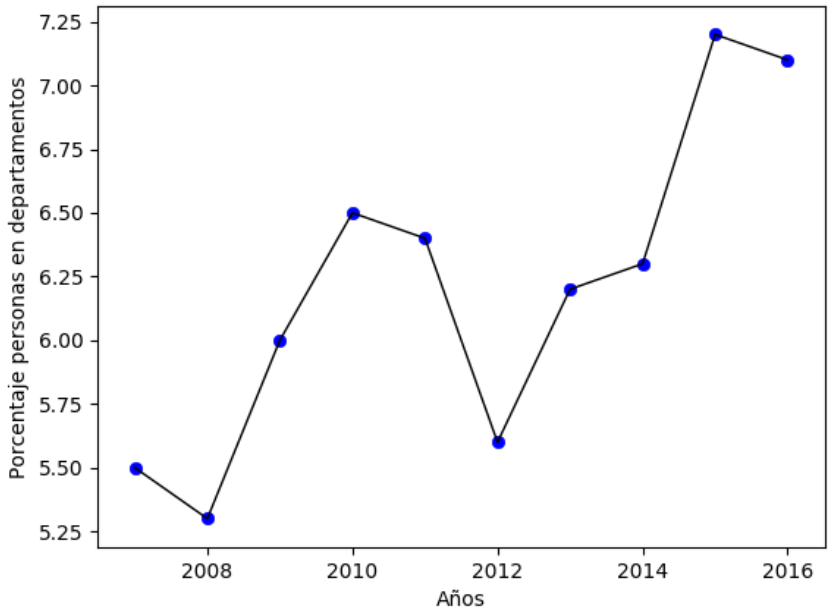


Figura: Graphics of tenants

Data from INEI and SUNARP

León et al

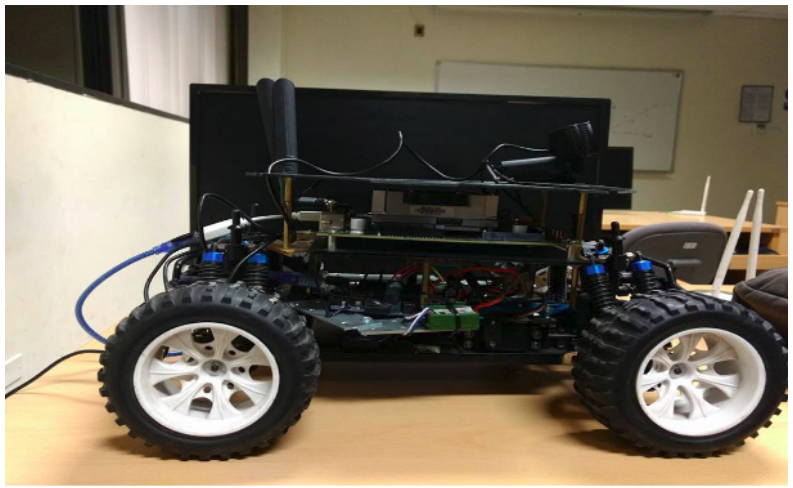
- We construct a database with all information about the tenants.

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	...

(a) DataBase

Car monitoring
León et al

- Built using a base of a Monster Truck 1/10 with a motor and 2 servos for the movement of front and rear tires.
- An Arduino UNO communicate with the NVIDIA Jetson TX1 board.



(b) Vehicle

Model
León et al

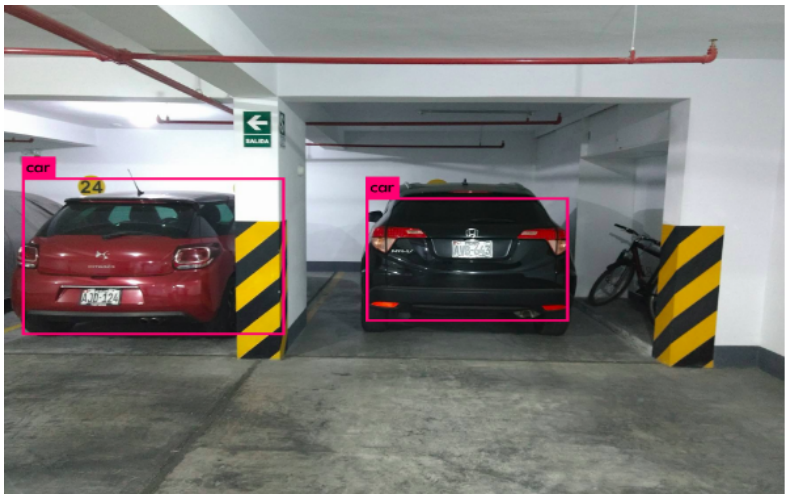
- Convolutional NN based on NVIDIA (End to End for self-driving cars).
- Conv24-Conv36-Dropout-Conv48-Dropout-Conv64-Dropout-Conv64-Dropout-FC-FC-FC-FC with non - linear activation function ReLU.
- NVIDIA P4000 with a partition of 20 % validation set and 80 % training set and the ADAM optimizer method.

Vehicle and Path Detection

León et al

- NN 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.
- NN test: NVIDIA Jetson TX1 board with a C920 camera obtaining 15 fps which works well with the tiny YOLO model.

(c) Cars Recognition



(d) Path Recognition

Licenses plate recognition
León et al

- OpenALPR library has been tried to detect the plate license of each car and later to compare with SUNARP dataset.

(e) SUNARP FORM

Calculating position

León et al

- To calculate the position, we use Beacons and we triangular position.

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

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

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

Calculating:

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

We obtain:

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

By which we would have:

$$x_c(x_i - x_j) + y_c(y_i - y_j) = Y_i, \text{ AX = Y}$$

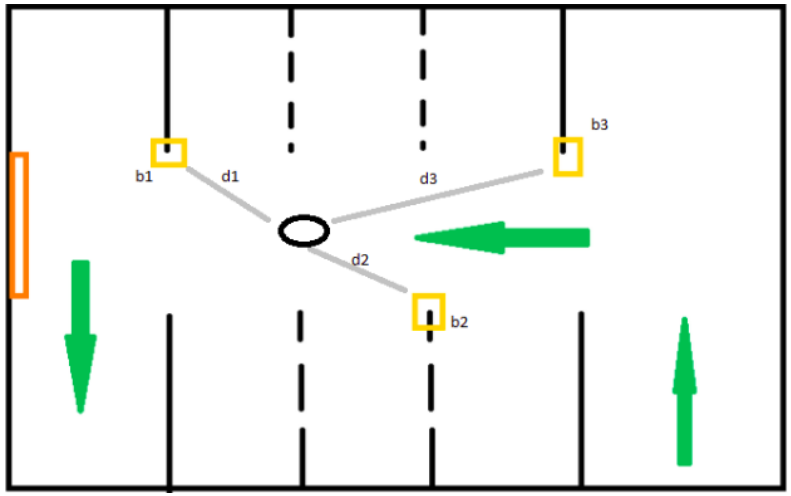
Calculating position
León et al

Where:

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

A: is the matrix forms by row vectors.

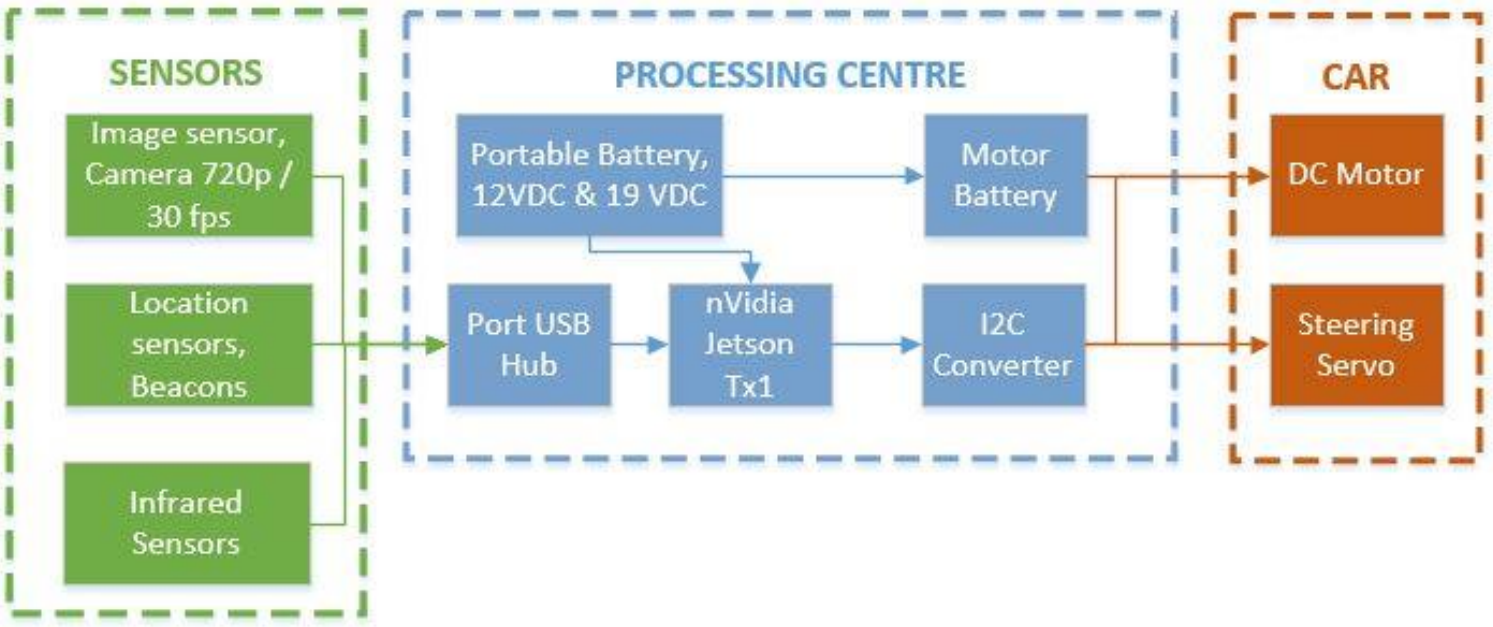
$$A = \begin{bmatrix} x_1 - x_2 & y_1 - y_2 \\ x_2 - x_3 & y_2 - y_3 \\ x_3 - x_1 & y_3 - y_1 \end{bmatrix}$$



(f) Triangular position

System structure

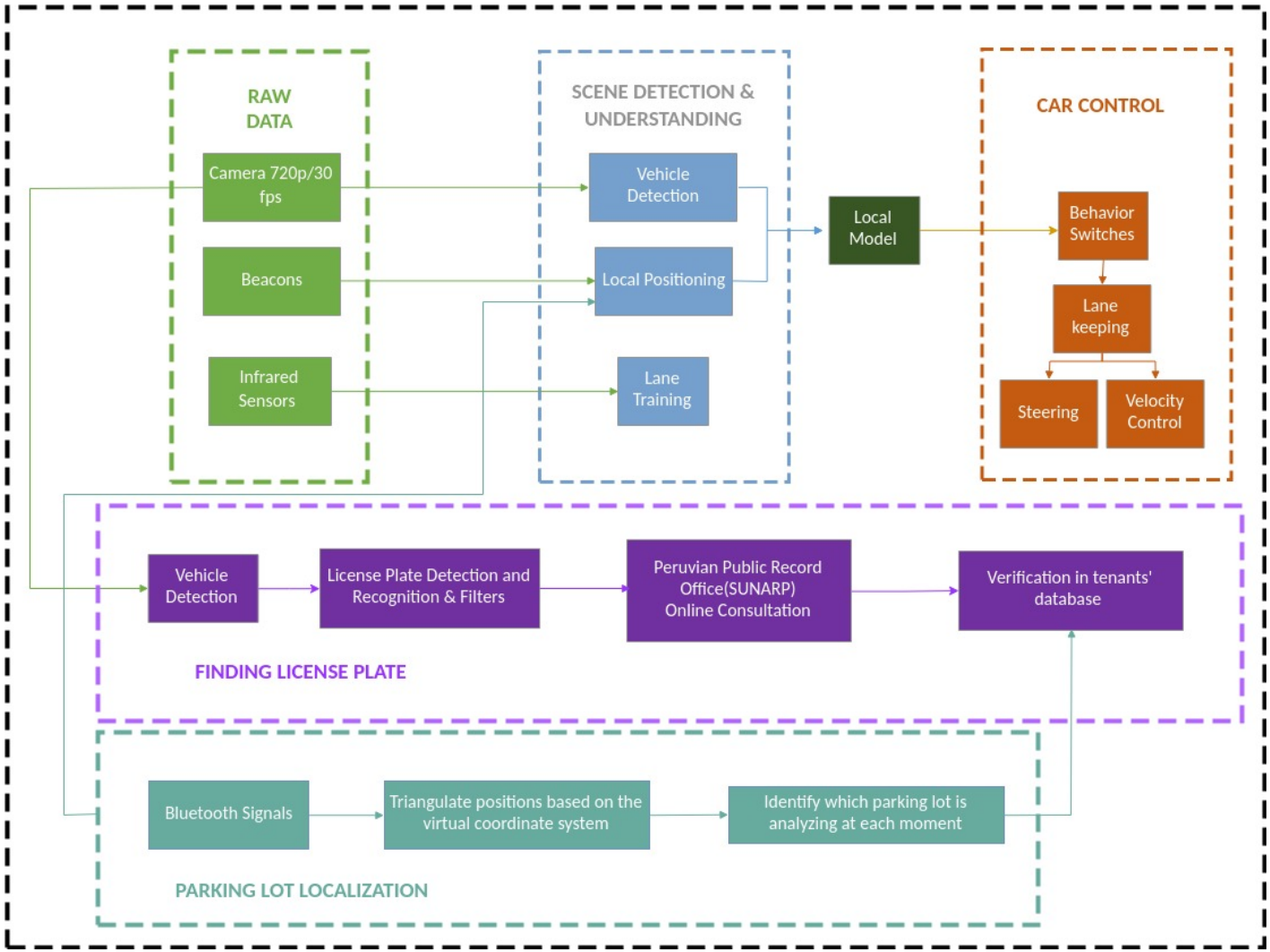
León et al



(g) Hardware Communication

Hardware Communication

León et al



(h) System Structure

Gracias!