Turkish Journal of Computer and Mathematics Education *Vol.12 No.6 (2021), 4470-4477*
Research Article

A Smart and End to End approach for Detecting free Parking Lots Using Neural Network

Shuvam Bhattacharjee, Ritik Khandelwal, Kanishk Malik, Sellam. V

Student, SRM Institute of Science and Technology, Ramapuram Campus, Chennai [1, 2, 3]
Assistant Professor, Department of Computer Science and Engineering [4]

Abstract - Parking a vehicle is one the most difficult and tedious tasks that annoys most of us on a daily basis. Our aim is to make a system that helps us to make this task to automate and help us find a parking spot as near as possible. In this paper we aim to find a parking spot in a city so that users can while traveling to their destination can find a parking spot near to their destination and can book it just before they arrive. Our system works using a neural network and lives image dataset so that it works in real time. And we can present our output to the user in their phone so that they can check and choose their parking spot. We aim to use private, public and roadside parking spots in this paper.

Keywords - Convolutional Neural Network (CNN), Image Processing, Smart Parking System, Long Short Term Memory (LSTM).

I. INTRODUCTION

Finding an empty parking spot is quite a difficult task and makes it more difficult during busy hours. On average one person drives extra 30 kilometers to find a parking spot every year. This not results in more money and fuel wasted but also more pollution. According to another study it found that every driver drives more than 8 minutes every day finding a spot and this process is very fuel consuming and not efficient at all.[1]

As of late, many thinkers and technologists are trying to better the parking lot vehicle detection system, few of them use simple photo observations and detections. which includes constant translation of pictures which assists us to computerizing the identification of parking spots. While others keep record of vehicle movement so that they can detect which parking spot is free and which one is occupied. There are two significant methodologies for distinguishing parking spaces. Another way is to fit a parking lot with sensors to detect the presence of a vehicle. This is the old method but very efficient and cost effective, while image detection is the new method and can solve the problems of the future.

So, this paper will use the first approach to find and serve users with appotiapoate parking spots closest to their destination, using live images taken by various cameras across the city. Then these images will go to a system to find whether the spot is empty or not and the user and non user can park depending upon their preferences. The users can book the parking spot just before their arrival so that the system can track the user just like google maps and give the spot to the user and he doesn't have to find the spot and save their time and fuel. But we have to keep in mind that booking a spot more than one minute to the user is not an option even to the extra paying user since it is injustice to others especially to non users.

So, this paper tries to make this process very efficient such that a user can input the destination in their mobile phone and the system can find a spot nearest to their destination and book it one minute before so that some other person doesn't have to suffer due to this. We aim to use all the possible parking spots in a city such as public, personal(rental) and roadside parking in this paper. In the system we will classify some parking spots as electric vehicle(EV) spots such that electric vehicles can park in that spot and can recharge their vehicle, as we believe that electric vehicles are the future car type rather than combustion engine type. [2]

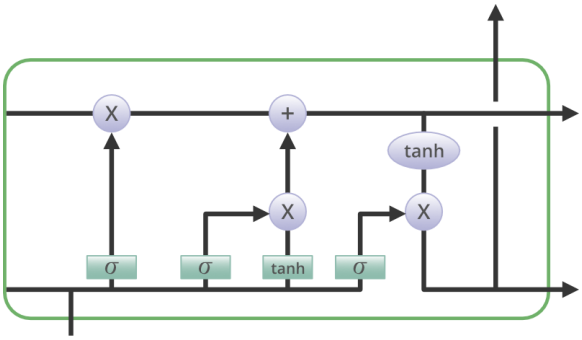

Figure 1: Structure of LSTM

II. Related Work

The existing system provides us two main aims for this project, first of which is to minimise the parking expenses and the other one is to balance the load of parking in various parking lots. Then the study of both factors needs to be considered and find a right balance between them formulating a mixed integer linear programming problem. Various other papers tied to solve it using CNN [2, 4, 5] or graph based solutions [7] or various sensor related solutions. We aim to solve the problem using an alternating direction method of multipliers (ADMM)- based algorithm that is a distributed implementation.

System overview

The project aims to use some frames from pre-installed cameras in parking lots and determine whether the slot is empty or not and update the system every 1-2 minutes. The second part of this project is to show the data to our users most preferably in an mobile phone application so that they could know whether a parking place is empty in an area or not and straight can drive to the parking slot. So the second part needs a server mainly to solve four responsibilities, to take care of the database that is gathering from the cameras. This connects the front-end system with the database to present data, and it receives the image data from visual nodes to the detection module. This module is the brain of the system that detects the appropriate output for the system by converting a lot is empty or not. [3]

Pre-processing

The raw data that is the images is to be taken from cctv cameras to detect edges and to recognize whether the parking lot is empty or not. So the first step in video to image conversion is to take some frames from raw video and these frames have to be converted to low gray level values. We do this process because it makes the system more efficient and faster. This has to be followed by cancelling the hue and saturation information while retaining the luminance. And the log transformations whose job is to map the narrow range in our step one of gray-level value. In the input values so that we can control the levels in our output levels. Second step includes derivatives giving a strong reaction to fine details as fine lines. The third step is binary transformation, where it indicates a digital image of two binaries as 0 and 1 where each represents black and white respectively according to luminance. Edge detection is a fine line between mathematical methods and train images which are aimed to train it digitally[4]. This has become the fundamental tool of the modern generations as this can solve the problems of the new generations and solve them in feature detection and feature extraction. [5]

III. SYSTEM MODULES

A. CNN

In this model, the cost of prediction was made very less, but there was an issue for prediction results changes at some moments. We think and prove in this paper that the NN algorithm is more helpful and better in every aspect, which is excellent for non- linear complex problems. On the contrary, prediction based on the stochastic probability model can restrain the deviation in prediction very well, that is the main and big drawback of NN in a medium and longer period of time and prediction. [6]

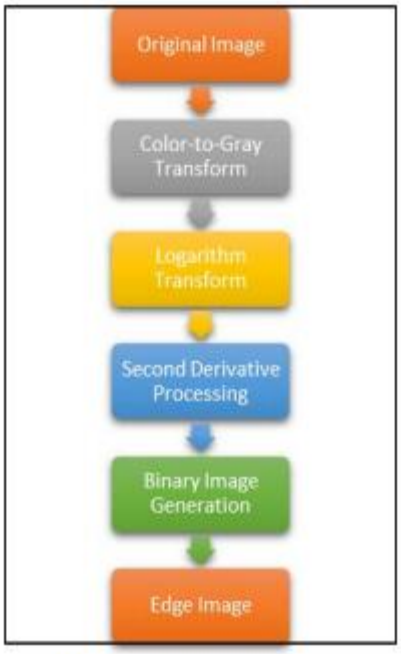

Figure 2: Preprocessing stages

The starting point of a model's construction is to make a probability model based on some real stuff and samples. And in starting for prediction, it is important to verify the model's ordinary conveyance of stopping arrangement in each time cut. After the trial of ordinariness, we figure the dispersion boundaries by the typical dissemination fitting capacity of normfit. In this way, the stochastic arrangements are produced dependent on these appropriation boundaries.[7] Stopping the request age subarea ought to be controlled to a sensible size. On the off chance that the subarea is too enormous, the accuracy of stopping seats would be weakened. On the off chance that it is too little, the information overview and handling work strength would be expanded.

It takes the hubs associated with the most limited way as the brief imprint focuses in the inquiry cycle. Each cycle looks through the hub with the most limited way from the source point as the indelible marker hub from the brief marker hub, until the objective hub is found or all hubs become indelible marker hubs to end the calculation.

B. Edge detection

Edge detection is an important part of the RNN network system to find out where our system can find any vehicle edges or not. Finding any edges we imply that there is a vehicle parked there. So the following formulas we used to find such.

$$\begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{df}{dx} \\ \frac{df}{dy} \end{bmatrix} \quad (1)$$

Among the vital highlights of a picture for example lines, edges, and focuses, we have utilized edges in our current work which could be distinguished from the unexpected changes in the dark level. An edge basically outlines between two unmistakably various districts, which implies that an edge is the boundary between two various locales. [8]

The magnitude of the gradient is given by

$$G[f(x,y)]=\sqrt{Gx^2+Gy^2} \quad (2)$$

The direction of the gradient is

$$B(z,y)+\tan^{-'}(G_y\sqrt{G}) \quad (3)$$

The adjustment in power level is estimated by the angle of the picture. Since a picture g (z, y) is a two dimensional capacity, its inclination is a vector. [9]

$$G_x=\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} *A \;\&\; G_y=\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} *A$$

(4)

Inclination administrators require two covers, one to acquire the X-bearing slope and the other to get the Y direction angle. These two slopes are consolidated to acquire a vector amount whose extent addresses the strength of the edge inclination at a point in the picture and whose point addresses the slope point. [9]

C. Long Short Term Memory

To assess the presentation of profound Long Short Term Memory network we utilized the root mean square mistake (RMSE), mean total mistake (MAE), mean squared blunder (MSE), middle outright mistake (MdAE), and mean squared log blunder (MSLE). The numerical definition of these presentation assessment strategies is characterized as:

$$\text{RMSE} = \sqrt{1/n\sum_{i=1}^{n}(y_i - \hat{y_i})^2} \quad (5)$$

$$\text{MAE} = 1/n\sum_{i=1}^{n}|y_i - \hat{y_i}| \quad (6)$$

$$\text{MSE} = 1/n\sum_{i=1}^{n}(y_i - \hat{y_i})^2 \quad (7)$$

$$\text{MdAE} = \text{median}\left(\sum_{i=1}^{n}|y_i - \hat{y_i}|\right) \quad (8)$$

$$\text{MSLE} = 1/n\sum_{i=1}^{n}\left(\log((y_i)+1) - \log((\hat{y_i})+1)\right)^2 \quad (9)$$

The image yi is the genuine profit parking spot that is figured by taking the distinction in inhabitants and limit esteems. Whereas, yiˆ is the anticipated accessible parking spot, anticipated by profound LSTM organization. The distinction between real accessible parking spot and anticipated parking spot is figured by yi − yiˆ. The exploratory outcomes accomplished utilizing a profound LSTM organization.[10]

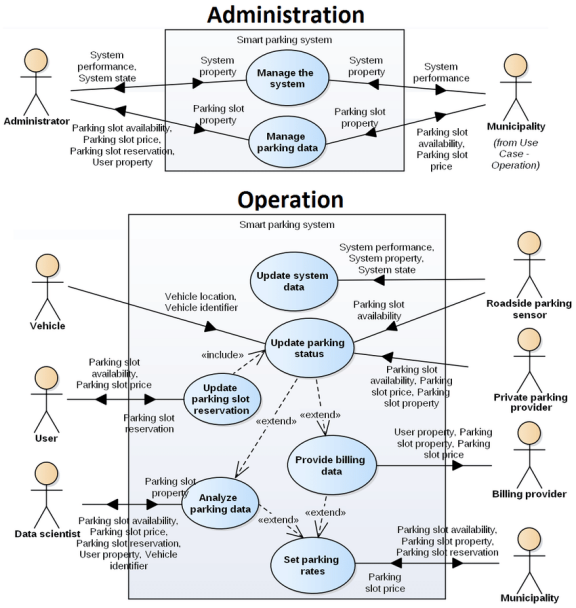

Figure 3. Smart Parking use case

IV. Analysis

The proposed framework was carried out by mimicking the framework practically. Even after this the best output and results was got by applying the Decision Tree calculation. We found choice trees are not extremely complex and time consuming, not at all like other such models because the main reason behind this is that it does not include any confounded conditions or

formulae . Subsequently, the Decision Tree calculation ends up being the most reasonable model and has been applied in the proposed framework . Various applications created in the framework have been tried utilizing and stopping the pictures and dealing outcomes were acquired. The client is given two alternatives – to see all empty stopping openings close by and furthermore then closest stopping space according to their area.

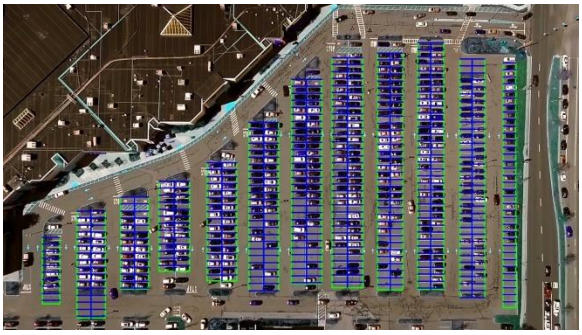

Figure 4. Parking spot detection

V. Result

The project has been trained using Pklot dataset which contains 29,000 odd images of parking lots around the world. We made a system that has tested on available various databases and have worked really well as per our expectation. Our recurrent neural network has performed and scored an accuracy of high 90%. The robustness of the system has also been tested and our model can work in real time data.

VI. Conclusion

In this paper, developed a parking lot detection system and have implemented in real-time so that users can find a parking spot across various private and public owned parking spots in a city. We achieved this system using an RNN classifier, this system has developed an end-to-end approach from front end application to serve the user with appropriate data. This system is now ready for practical use and can solve the problems of the future.

VII. Future Work

In future, the project can be integrated with google maps and other similar applications so that users can easily see their destination with a predefined parking spot. These projects can also be finely tuned so that we get 100% accurate results within seconds of real life implementation.

REFERENCES

[1] Tom Thomas, Tarun Bhatt "Smart Car Parking System using Convolutional Neural Network" Padre Conceição College of Engineering.

[2] Swasthi B S, Anagha R, Arpitha S, Sanjay B S, Harshitha K "Parking Assist using Convolution Neural Networks " Vidyavardhaka College of Engineering, Mysuru.

[3] Archana Dehankar, Amit A. Kamble, "Review on Automatic Car Parking Indicator System" IJRITCC, Sep 16, 2015.

[4] Weilin Yan and Kourosh Khoshellam,DebadityaAcharya, "Real-time image based parking Occupancy detection using deep learning", University of Melbourne.

[5] C A Gopinath,J Cynthia, Bharathi Priya, "IoT based smart parking management system", IJRTE ,2018.

[6]PratikshaJagtap,SnehalChoudhary,NazishFatima,AkshayaNatkar,"IoTbasedsmartparkingsystem for smart cities", IJARIIT.

[7] J Ganesh,C V Sukhesh,M Mohan,E Cassin Thangam, "Internet of Things(IoT) based Smart parking reservation system using Raspberry-pi", IJAER.

[8] Gurobi Optimization The Best Mathematical Programming Solver. Accessed, 2019.

[9] H. Choi, and S. V. Dhople, P. J. Seiler, "Propagating uncertainty in power flow with the alternating direction method of multipliers," IEEE Trans. Power Syst., Jul. 2018.

[10] W. Yin and W. Deng "On the global and linear convergence of the generalized alternating direction method of multipliers," J. Sci. Comput., Mar. 2016.

.