JOHANNES KEPLER
UNIVERSITY LINZ

Author
Markus Hiesmair

Submission
Institute of Telecooperation

Thesis Supervisor
Karin Anna Hummel

February 2018

ROBUST DRIVE-BY ROAD SIDE PARKING DETECTION ON MULTI-LANE STREETS USING AN OPTICAL DISTANCE SENSOR

Master's Thesis

to confer the academic degree of

Master of Science

in the Master's Program

Computer Science

JOHANNES KEPLER
UNIVERSITY LINZ
Altenberger Str. 69
4040 Linz, Austria
www.jku.at
DVR 0093696

Affidavit

I hereby declare that the following dissertation ”Put your thesis title here” has been written only by the undersigned and without any assistance from third parties.

Furthermore, I confirm that no sources have been used in the preparation of this thesis other than those indicated in the thesis itself.

Linz, on November 9, 2017

Markus Hiesmair

Acknowledgment

Summary

Summary ...

asdf

Abstract

Abstract ...

Contents

To-Do: For now the TOC depth is 5 but will be reduced later

Abbreviations

GPS Global Positioning System

LIDAR Light detection and ranging

List of Figures

List of Tables

Chapter 1

Introduction

Traffic congestion in urban areas became a bigger problem every year in the past decades. Increasing traffic causes several issues, for example high monetary and environmental costs by using gasoline and by emitting CO_2 to the environment. There are several strategies to reduce urban traffic to mitigate these problems like for example new investments in public transport infrastructure. However, the usage of private cars in cities won't stop in the next decades, but increase. According to a study by Hao et al. [2] car sales will continue to grow in the upcoming decades and therefore also traffic will continue to become worse in big cities. Thus, it is important to find ways to reduce traffic in urban areas.

With more vehicles driving in urban areas, there also comes the need for a sufficient number of parking spaces. Finding parking spaces in urban areas can be a really difficult, frustrating and time consuming task for drivers. There often exists some information about the availability of parking spaces in parking garages, but in most cities the situation of road side parking is rather non-transparent. This not only leads to frustrated drivers, who are searching for parking spaces a long time, but again contributes to urban traffic congestion as many cars have to drive around close to their destination while searching for free parking spaces.

There are many studies, which show that the searching for parking spaces adds a lot of traffic. In 2013 a study by Nawaz et al. [9] showed that about 30% of traffic congestion is created by drivers looking for free parking spaces. Another study [3] found that alone in 2007 searching for parking spaces caused costs of about 78 billion US dollars by using 2.9 billion gallons of wasted gasoline and 4.2 billion lost hours only in the United States. Furthermore, this obviously causes a lot of CO_2 emissions which is not only bad for the environment and contributes to climate change, but also lowers the quality of living in big cities through the significant amount of air pollution.

One of the most important contributors to high search times for parking spaces is not only the lack of vacant parking spaces, but also the lack of information, if and where free parking spaces are available. Therefore, one way to mitigate many of the above stated problems is to determine the current parking space situation in the city and make it accessible to the public (e.g. via web application), so that drivers can efficiently navigate to a vacant parking space, or even decide if they want to go by car or use public transportation, depending on the number of parking spaces available close to their destination.

However, detection of road side parking spaces and their states is a challenging task. Of course an obvious approach to the problem would be to put stationary sensors to every parking space in the city, which check, if the corresponding parking space is occupied or vacant. This, however, has the drawback to be very expensive as, for big cities, thousands of sensors would have to be bought, installed and maintained. Furthermore, because the state of parking parking spaces does not often change, the high frequency of sensing with such a system would be rather inefficient.

1.1 Drive-By Park Sensing

A promising novel approach to sense a city's parking situation is the use of mobile sensors instead of static ones. Crowd sensing has the advantage to be usually more cost effective and can provide sufficient accuracy for the purpose of providing parking space availability maps.

There are several approaches to mobile parking availability sensing, which will be discussed in chapter 2. This thesis will focus on "drive-by park sensing". The idea behind the drive-by sensing approach is that there are sensing vehicles which are driving through the city and collecting data of their environment through mounted sensors. Using the collected data parking cars and vacant parking spaces should be detected. There already exists a prototype implementation of such a system. In 2013, Mathur et al. [7] presented their system, called ParkNet, which continuously measured the distance to the nearest obstacle on the right side of the road, as well as the location of the sensing vehicle through a GPS sensor. Using these data, they used thresholding to detect parking cars and vacant parking spaces. A more detailed description of their work is being described in section 2.1.5.1.

Figure 1.1 shows a standard drive-by scenario of a sensing vehicle which passes two parallel parked cars and a vacant parking space in between them. The distance measurements while passing the parked cars will be much shorter than the measurements

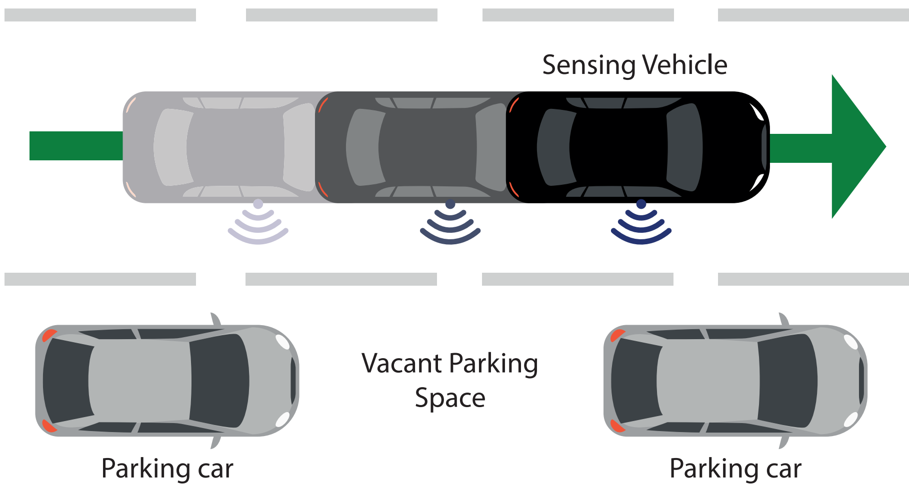

Figure 1.1: The sensing vehicle passes two parked cars and should identify a vacant spot in between using distance and location measurements. **To-Do:** Hier Referenz zu Parknet-Paper??? Ähnliche Grafik...

taken while passing the vacant parking space. This should allow a basic algorithm to recognize parking cars and vacant parking spaces.

However, the situation which is shown in figure 1.1 is an idealistic one. In real life traffic, there will be much more complex situations to face, which are not as easily detectable and which might influence the success of the detection. For instance, the sensing vehicle might not drive in the right-most lane, therefore the measured distances will be much longer. Another possible issue are other driving cars, motorcycles or bicycles which the sensing car overtakes. There are many of such distractions which have to be filtered out to ensure that there are as few false detections as possible. The existing ParkNet prototype does not take any of such situations into account, because it was only evaluated on parallel parking cars on single lane roads.

The high complexity of urban traffic and the many distractions during sensing make it nearly impossible to create a rule set based on the sensor measurements which would be able to detect parking situations at a sufficient accuracy. Therefore, simple thresholding as applied in the ParkNet prototype will not work in real life traffic scenarios. Furthermore, such rule sets would have to be created for each city individually because of the different nature of the roads and the parking spaces. For instance, the distance between roads and parking spaces can vary highly in two different cities. Thus, to be able to detect parking situations, new methods have to be found, which should be more flexible to distractions and changes in the environment.

1.2 Research Goals and Methodology

The overall goal of this thesis is to determine which accuracy can be achieved for detecting the parking space situation in urban real world traffic scenarios using drive-by park sensing. Furthermore, distracting situations should be identified and strategies to cope with them should be found to be able to keep the detection accuracy high. In contrast to existing approaches in this thesis machine learning will be applied to classify different road traffic and parking scenarios to be able to better cope with the high variety of situations possible. As a first step, sensor data has to be collected in real world traffic scenarios including distance sensor measurements and GPS locations of the sensing vehicle. To be able to apply supervised learning all sensor measurements have to get assigned their ground truth manually. As next step different machine learning algorithms will be applied, evaluated and compared to each other in terms of accuracy for the different classes. The detailed methodology which will be followed includes the following steps:

Building a test bed As a first step a test bed has to be built, which is able to access the required sensors to record all the necessary data. A Raspberry Pi will be used as processing device because of its popularity and the many compatible sensors which work with this platform. It is connected to a LIDAR-Lite v3 sensor which continuously measures the distance to the nearest obstacle on the right side of the road. A GPS receiver will track the location of the sensing vehicle and a camera will be used to record images of the ground truth for evaluation purposes only. In section 3.1 the complete setup of the test bed is described as well as all the specific hard ware parts and their abilities.

Acquiring a dataset As soon as the test bed is ready, the sensors should be mounted on the prototype car to be able to start recording the dataset. Test drives should be done in some selected streets in Linz, Austria with the focus of variety of the recorded situations. The test scenes should include single lane as well as multi lane streets and measurements in all streets should be done several times. Furthermore, the car should be driving as it would in regular traffic (not only in the right most lane, etc...) and the scenes should also include high and low traffic scenarios to have a high amount of diverse data. All measured distances, GPS locations and ground truth images have to be saved to files with the according timestamps to be able to evaluate the results of different approaches later on. Furthermore, using the images taken by the camera, ground truth values will be manually labelled in different classes (Parked car, free space, overtaken car, ...). A detailed description about the dataset and the ground truth tagging can be found in section **To-Do:** ref **??**.

Data processing and segmentation As next step the measured sensor values have to be preprocessed and filtered in order for the following algorithms to work. Sensing and overflow errors as well as outliers in the measurements should be identified and removed before further processing. After the raw data has been filtered, the sensor data has to be segmented. As parking cars and other cases which should be classified consist of several sensor measurements, the corresponding sensor measurements should be grouped together and merged to segments which will be later classified. All preprocessing steps and the segmentation process are described in more detail in section **To-Do:** ref 3.3.

Classification using basic machine learning techniques Features on the created segments are calculated (for instance length, average distance and variation of the distances) and are being used to train and evaluate several machine learning algorithms which will be compared on their performance. Furthermore, some deep learning models will also be evaluated on the raw sensor data of the segments and will be compared to common machine learning results. The results of all experiments can be found in section **To-Do:** ref **??**.

Further improvements **To-Do:** todo

Chapter 2

Related Work

This chapter will cover work related to park sensing and machine learning. Section 2.1 will discuss different approaches to sensing current parking situations in cities. Furthermore, a comparison of them will be given as well as advantages and disadvantages of the specific approaches. **To-Do:** Machine Learning...

2.1 Approaches to Park Sensing

There already exist numerous approaches to detecting the states of parking spaces as well as the parking situation in a city. In this section parking detection approaches will be categorized in five different categories and in the following subsections several reference papers for all categories will be discussed. The first section about stationary park sensing (section 2.1.1) will reference approaches where sensors are stationary deployed per parking space or parking area. Section 2.1.2 will discuss counting in- and outgoing vehicles to coarsely detect parking space counts. Another approach is to detect certain events (for instance parking and unparking) to estimate parking space counts. Section 2.1.3 will discuss several related papers in this area. Drive-by sensing will be discussed in sections 2.1.4 and 2.1.5. Using cameras to detect parking spaces while driving by will be referenced in section 2.1.4, while section 2.1.5 will cover using distance sensors. **To-Do:** evtl. comparison, advantages, disadvantages section

2.1.1 Stationary Park Sensing

The most obvious and technically simple solution to park sensing is to use stationary sensors to determine the state of parking spaces. Usually one sensor per parking space is used which determines its state and sends it to a central server. However, some approaches exist where one sensor (a camera) senses several parking spaces close to each other. Reference projects of both solutions will be discussed in this section.

In San Francisco a first prototype of the SFpark project [10] has been implemented from 2008 to 2011. In specific down town areas in San Francisco with high amounts of traffic, wireless stationary sensors have been placed at about 8.200 road side parking spaces. These sensors are able to detect the state of one parking space in real time and send this information to a central server. Furthermore, parking garages also count the in- and outgoing vehicles and shared this information, so that the parking situation in the areas with park sensing can be derived. The gained information is being shared as open data, so third party developers and researchers can also use the dataset for all kind of projects. Furthermore, there exists an App from SFpark itself to help drivers find available nearby parking spaces, navigate to it and pay as they go with their phones.

A top priority goal of the SFpark project is to increase the availability of parking spaces in every block throughout the city. To achieve this goal, they are using demand responsive pricing. If (almost) all parking spaces in a neighbourhood are occupied for a long time, they raise the price in this specific area and vice versa if no parking spaces are occupied they lower prices. This leads to high overall parking space availabilities (20 - 40%) and also to lower traffic congestion and CO_2 emissions. However, besides the advantages of high accuracy and being a real time system, there also are disadvantages. First of all, only metered parking spaces can be tracked, as sensors have to be installed per parking space, so areas where parking is allowed but there are no clearly marked parking spaces cannot be sensed with a reasonable accuracy. Furthermore, another big drawback are the high overall costs. The about 8.200 stationary sensors have to be bought, installed and maintained, which obviously causes high costs while only covering a tiny fraction of the overall San Francisco down town. So if the system should be available in the whole down town, costs would increase dramatically.

There also exist a lot of similar projects which deploy stationary sensors at a high range.

London, Streetline, Google Open Spot

Advantages: Highly accurate. Real-time.

Disadvantages: Highly inefficient. Only metered parking spaces. High costs.

multi-classifier image detection system[5]

arm smart camera [4]

2.1.2 Counting Vehicles

smart urban parking detection [11]

2.1.3 Event Detection based Park Sensing using Smartphones

updetection[6]

parsense [9]

pocketparker [8]

2.1.4 Drive-by Park Sensing using Cameras

2.1.4.1 ParkMaster

parkmaster [1]

2.1.5 Drive-by Park Sensing using Distance Sensors

2.1.5.1 ParkNet

parknet... [7]

2.2 Machine Learning

Chapter 3

Prototype Implementation

blabblabla

3.1 System Design

- System description (type of sensors, requirements); overall envisioned system of cars detecting free spaces and communicating the information - Testbed description (concrete HW and technical capabilities)

3.1.1 Test Bed Description

3.2 Experiment description and data collection

- Description of experiments (scenarios we are interested in: parking car, etc.) - Data set derived (raw data, size, etc.) - Map data and camera ground truth data

3.3 Data processing

- Preprocessing, feature definition ... - Traditional ML Algorithms (incl. a short description of each algorithm and configuration options) - Deep Learning

3.4 Experimental results

- Results of the different ML approaches, important features, difficult cases, etc. - Comparison of traditional ML and Deep Learning (cf. our demo)

Chapter 4

Results and Discussion

Chapter 5

Conclusions and Future Work

Bibliography

[1] Giulio Grassi, Kyle Jamieson, Paramvir Bahl, and Giovanni Pau. Parkmaster: An in-vehicle, edge-based video analytics service for detecting open parking spaces in urban environments. In *Proceedings of the Second ACM/IEEE Symposium on Edge Computing*, SEC '17, pages 16:1–16:14, New York, NY, USA, 2017. ACM.

[2] Han Hao, Yong Geng, and Joseph Sarkis. Carbon footprint of global passenger cars: Scenarios through 2050. *Energy*, 101(Supplement C):121 – 131, 2016.

[3] Texas Transportation Institute. Urban mobility report. 2007.

[4] Y. Li, G. Li, and X. Zhao. Approach for parking spaces detection based on arm embedded system. In *2013 5th International Conference on Intelligent Human-Machine Systems and Cybernetics*, volume 2, pages 406–409, Aug 2013.

[5] J. Liu, M. Mohandes, and M. Deriche. A multi-classifier image based vacant parking detection system. In *2013 IEEE 20th International Conference on Electronics, Circuits, and Systems (ICECS)*, pages 933–936, Dec 2013.

[6] Shuo Ma, Ouri Wolfson, and Bo Xu. Updetector: Sensing parking/unparking activities using smartphones. In *Proceedings of the 7th ACM SIGSPATIAL International Workshop on Computational Transportation Science*, IWCTS '14, pages 76–85, New York, NY, USA, 2014. ACM.

[7] Suhas Mathur, Tong Jin, Nikhil Kasturirangan, Janani Chandrasekaran, Wenzhi Xue, Marco Gruteser, and Wade Trappe. Parknet: Drive-by sensing of road-side parking statistics. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, MobiSys '10, pages 123–136, New York, NY, USA, 2010. ACM.

[8] Anandatirtha Nandugudi, Taeyeon Ki, Carl Nuessle, and Geoffrey Challen. Pocketparker: Pocketsourcing parking lot availability. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '14, pages 963–973, New York, NY, USA, 2014. ACM.

[9] Sarfraz Nawaz, Christos Efstratiou, and Cecilia Mascolo. Parksense: A smartphone based sensing system for on-street parking. In *Proceedings of the 19th Annual International Conference on Mobile Computing & Networking*, MobiCom '13, pages 75–86, New York, NY, USA, 2013. ACM.

[10] SFMTA. http://sfpark.org. 2017.

[11] N. R. N. Zadeh and J. C. D. Cruz. Smart urban parking detection system. In *2016 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, pages 370–373, Nov 2016.