Drive-by Sensing for On-Street Parking Spot Detection

by

Yuxuan Zhang

B.Eng. in Computer Science and Technology, Huazhong University of Science and Technology, 2022

Submitted to the Graduate Faculty of

the Department of Computer Science in partial fulfillment

of the requirements for the degree of

Master of Sciences

University of Pittsburgh

2024

UNIVERSITY OF PITTSBURGH

SCHOOL OF COMPUTING AND INFORMATION

DEPARTMENT OF COMPUTER SCIENCE

This thesis was presented

by

Yuxuan Zhang

It was defended on

March 26, 2024

and approved by

Dr.Longfei Shangguan, School of Computing and Information

Dr.Xiaowei Jia, School of Computing and Information

Dr.Stephen Lee, School of Computing and Information

Drive-by Sensing for On-Street Parking Spot Detection

Yuxuan Zhang, M.S.

University of Pittsburgh, 2024

On-street parking remains a persistent challenge, leading to driver frustration and wasted time. More critically, drivers circling streets in search of parking contribute to traffic congestion, increased carbon emissions, and unnecessary fuel consumption. The crux of the on-street parking dilemma lies in the insufficient awareness of available parking spaces. To tackle this issue, various solutions, including static and mobile sensing methods, have been explored and implemented. Yet, these strategies have encountered obstacles that hinder widespread adoption. Static sensors, for instance, are typically limited to monitoring a single parking space each, leading to high costs for comprehensive coverage. Mobile sensing strategies, on the other hand, aim to maximize sensor utility by collecting data on multiple spaces. However, these methods have traditionally required specialized hardware installations on vehicles, posing barriers to large-scale application.

In this thesis, we introduce an innovative passive mobile sensing solution that mitigates the need for dedicated hardware installation. Our key observation is that moving vehicles inherently emit signals, predominantly in the form of tire noise and aerodynamic noise. When parked cars are present along the roadside, these sounds reflect back to the moving vehicle. By leveraging a smartphone to capture these naturally generated signals, we can effectively differentiate between empty spaces and parked cars. To realize this idea, we have developed an end-to-end system that achieves equal performance with the state-of-art mobile sensing technologies in detecting available on-street parking spots. Our system comprises a pre-processing module that employs signal processing techniques for automatic data segmentation, a Deep Neural Network (DNN) model for parking spot availability prediction, and a post-processing component that refines the parking information. To support our model, we created our own dataset from data collected over 2 weeks, including 2,512 samples in total. Our approach not only demonstrates the feasibility of hardware-free, software-based solutions for on-street parking spot detection but also holds the potential for scalable imple-

mentation across urban environments.

Keywords: passive mobile sensing, machine learning, signal processing, on-street parking spot detection.

Table of Contents

List of Tables

List of Figures

Preface

I would like to take this opportunity to thank all the people who have provided me help and support in my pursuit of my master's degree.

Special thanks to Dr. Longfei Shangguan who have guided me in my research and navigated the direction of this thesis. Many thanks to Dr. Xiaowei Jia and Dr. Stephen Lee for serving as my defense committee, and for all the insightful advises they have provided. Thanks to Zhiwei Ren for the help with machine learning. Thanks to Haoyu Li for helping me with data processing. Thanks to Yongjie Yang and Chonghao Qiu for the help with the experiments. Thanks to Tao Chen for the advises on this project. It is truly a honor for me to work with Dr. Longfei Shangguan and all the excellent team members in the group. I had a cherishable memory working with you guys.

Among all the people in my life, I would like to give special thanks to my parents, who have provided me all the support I could possibly get in my entire life. It is your mental support that give me strength to carry on and stay brave in front of difficulties.

1.0 Introduction

Finding on-street parking spots has been a long-lasting problem globally [11, 13]. Drivers often find themselves driving in circles, seeking vacant spots, resulting in frustration, wasted time, and increased stress [12]. The quest for parking spot also contributes to traffic congestion, safety hazards, and economic setbacks [14]. For instance, in a study conducted in Los Angeles, it was found that drivers in the city spend an average of 85 hours per year searching for parking spots. This translates to a total of 47,000 wasted hours and 17 million gallons of wasted fuel each year, costing the city approximately 730 million annually[43].

Providing drivers with real-time parking information has proven to be an effective way to tackle these issues[26]. Currently there are two groups of works on detecting the parking availability. The first group of works *deploy fixed sensors* [32, 37, 23, 9] on the road to detect the occupancy and report the results to a server for user access. This approach is proven to be accurate and responsive. However, it comes with a significant infrastructure cost, making it less scalable in practice. As an example, consider the SF-Park project [42], which required an investment of millions of dollars for the installation of 8,000 smart meters for occupancy detection. Furthermore, there are additional hidden costs associated with these sensors, such as relay nodes for data transmission and other necessary components, which elevated the total expenditure to a substantial 24.75 million USD. Such high cost demands long term investment from the society and is hard to deploy in large scale.

Another group of works proposes *drive-by sensing* to detect on-street parking spot availability [30, 16, 10]. The basic idea is to take the cruiser as a mobile sensor to detect the occupancy as the car passes by these parking spots. Compared to the fixed sensor-based approach, this drive-by sensing approach is promising due to its reduced infrastructure cost, enhanced accessibility, and higher flexibility and scalability. For instance, ParkNet [30] recruits a vehicle fleet with each vehicle equipped with an ultrasonic rangefinder facing the passenger side to determine parking spot occupancy. Likewise, ParkMaster [16] adopts smartphone cameras to detect the parking spot occupancy as the car drives by. Despite their convenience and low-cost, existing drive-by sensing systems all require dedicated deployment of sensors (e.g., ultrasound-based range-finder) and suffer from certain shortcomings. The use of ultrasonic sensors cast extra energy cost to send out the ultrasonic sounds. Also, the fact that a ultrasonic sensor is required to be equipped on the passenger-side door may prevent this solution from broad adaptation. Although promising results are demonstrated in ParkMaster during daytime, the performance under low light environment remains uncertain (e.g. at night). And the inherent security concern caused by camera could be another barrier to promotion.

In this thesis, we present our solution of mobile sensing base on our observation in daily life: *when driving pass the road, there exists human-distinguishable differences between road-side parked cars and vacant road-side parking spots.* This sound is generally recognized as roadway noise, which comes from multiple sources, predominately in the form of tyre noise[2] and aerodynamic noise[27]. These inherently generated signal provides opportunity for vacant parking spot detection without proactively sending out signals from the cruising vehicle. Using a smartphone to collect the corresponding signals, we can gather parking availability information without additional dedicated-hardware requirements. To realize this idea, we first conducted an empirical study to deeply understand the source of the roadway noise. From our empirical study, we identified three major source of the roadway noise, with the frequency of 500-2kHz, 7.5k-8.5kHz, 16.5k-17.5kHz. The 500-2kHz signal comes from tyre noise while the other two signals come from aerodynamic noises. We focused on leveraging the inaudible band signals to detect available parking spots based on our quantitative analysis and privacy cosideration.

With the findings in our empirical study, we designed an end-to-end system to gather parking availability information through the audio data we collected from a smartphone microphone. Our system starts from a pre-processing module for data segmentation. We designed a signal processing pipeline to convert audio data stream into informative segments of 3 seconds. Then we designed a Deep Neural Network (DNN) model for vacant parking space prediction. To train our model, we built our own dataset through data collected over 2 weeks and consists of 2,512 samples from 10 hours of audio. Finally, our system consists a post-processing module which formalize the output of our DNN model and finalize the results that could be disseminated to the front-end user. Under our evaluation, our model achieves equal performance with the state-of-art mobile sensing solutions with 93.51% of prediction accuracy and 96.55% of True Positive Rate, the False Positive Rate is 19.78% with a Precision score of 0.9553.

In the following chapters, we present our work in the following structure: Chapter 2 provide a overview of existing solutions for parking availability information sensing. In Chapter 3, we present our empirical study that begins with the setup and then provide details of our findings and design choices. In Chapter 4, we first present an overview of our system, then we present design details of each module. After that, we present our evaluation in Chapter 5. We start from introducing our dataset, and present our results in the following sections. Finally, in Chapter 6 and 7, we conclude our work and point out possible future directions for following works.

2.0 Related Work

Parking information sensing relies on the use of sensors to gather real-time data regarding parking space availability. There are two main methods for collecting this information: stationary and mobile.

2.1 Stationary Sensing

In the stationary method, sensors are installed directly on parking spaces. These sensors can quickly detect the presence or absence of vehicles and provide updated information when the occupancy status changes.

Magnetometers are commonly deployed in stationary parking detection systems, especially for municipal applications. They measure changes in magnetic fields, which occur when large metal objects like vehicles are present. Magnetometers provide precise and easily interpretable signal patterns. For example, SFpark[41] in San Francisco that utilizes magnetometers can transmit 85% of events within 60 seconds on its large-scale roadside parking sensor network. However, they are generally limited to single-detection events and can be more costly than some other sensor types.

Passive infrared (PIR) sensors [24, 29] operate by detecting heat radiated from the human body or warm objects. These sensors are often employed collaboratively with other sensing technologies to ascertain if a driver has parked and exited their vehicle. Due to their reliance on heat emissions, PIR sensors are particularly effective at recognizing human presence, making them suitable for applications where it's important to differentiate between people and vehicles. However, they may not be as accurate in detecting non-human sources of heat, and their performance can be influenced by environmental conditions.

Active infrared sensors [33, 25] work by emitting an infrared beam and measuring the time it takes for the beam to bounce back after hitting an obstacle. They are commonly used to measure the distance to objects in front of them. These sensors are sensitive to environmental factors such as sunlight and other sources of infrared radiation, which can affect their accuracy. Despite this limitation, they are still valuable for proximity detection and obstacle avoidance in parking and other applications.

Optical sensors [6, 25] that detect the change in light must be installed where light can be obscured by a parked vehicle . However, optical sensors are vulnerable to any light source and transient staying objects, so their accuracy is still questionable. Inductive loops and piezoelectric sensors are contact-based sensors installed on road surfaces. Inductive loops [20, 8] are widely used for traffic surveillance and simply detect if a vehicle is passing over them. Piezoelectric sensors [49], on the other hand, can provide more detailed information based on the pressure exerted on them. Both types of contact sensors are susceptible to wear and tear due to frequent use and require intrusive installation.

RFID technology [52] is commonly integrated into smart parking payment solutions. Vehicles equipped with RFID tags can be detected by RFID readers installed on parking spaces. As the popularity of electronic toll collection (ETC) increases, RFID-based detection becomes more widespread. However, it is primarily useful for identifying vehicles with RFID tags and does not provide information on the occupancy status of parking spaces. Laser rangefinders [19] are often used to create 1/2/3D maps, especially for environmental perception and autonomous vehicle applications. Typically mounted on vehicles, they emit laser beams and calculate the time it takes for the beams to bounce back from objects. This data allows them to measure distances from different objects, including parked vehicles, aiding in parking space detection and navigation.

2.2 Mobile Sensing

2.2.1 Mobile Sensors

On the other hand, the mobile method utilizes the mobility of vehicles to collect parking information. Mobile sensors are attached to vehicles and can detect parking space occupancy as they pass by. Ultrasonic sensors[7, 30, 46] use sound waves to detect objects and measure distances. They are well-suited for outdoor environments and can provide a more complex signal pattern compared to some other sensors. Ultrasonic sensors are effective at detecting obstacles and objects within their range, making them useful for parking space occupancy detection. ParkNet[30] in San Francisco, for instance, relies on taxi cabs equipped with GPS receivers and ultrasonic sensors to collect data, with an average update interval of 25 minutes for 80% of the cells in the downtown area.

Cameras and acoustic sensors provide more complex and rich signal patterns. Cameras [3, 40, 45, 4] capture visual information that can be processed to detect vehicle presence and occupancy. Acoustic sensors [34] use sonar technology to detect objects and movement. Both of these sensors require advanced image and signal processing techniques to extract relevant information from the background noise. They have garnered research interest not only for parking applications but also for broader use cases such as security and environmental monitoring. QR codes [1] may also be used to help drivers identify and pay for parking spaces, with the system announcing the availability of spaces when a driver ends their parking session. However, QR codes alone cannot confirm if drivers have paid for their spaces or if they are occupied by vehicles.

2.2.2 Crowdsensing

One key concept in mobile parking information sensing is mobile crowdsensing, which utilizes smartphones to collect parking availability information from drivers themselves. Crowdsourcing has found practical application in certain smart parking solutions, particularly in the context of gathering real-time parking availability information from smartphone users. This concept has given rise to a new term known as "crowdsensing."

One common approach involves the development of smart parking applications that encourage users to voluntarily share information, a practice referred to as "participatory crowdsensing." For example, an application like ParkJam [21] exemplifies this approach. Researchers, such as Rinne et al. [38], have explored the advantages and disadvantages of mobile crowdsensing. Through mobile sensors, high-level conclusions can be drawn, including the availability of parking space in an area, the fullness of an area, or the determination that an area should no longer be considered full. Farkas and Lendak [15] conducted simulations of crowdsensing activities related to urban parking, using the MASON multi-agent simulation toolkit and visualizing the results on OpenStreetMap. They utilized five different scenarios to model individual behaviors, including parking times and probabilities. Interestingly, even drivers who do not actively participate in crowdsensing or employ decision support applications can benefit from these systems. Notable results were obtained in scenarios like Novi Sad, where 30% of drivers participated as crowd sensors, resulting in 14% shorter cruising times.

PhonePark [44] and Park Here! [39] introduced parking and unparking algorithms assisted by accelerometers, gyroscope sensors, and Bluetooth. Bluetooth aids in the detection of users' transportation modes. VeLoc [51] used smartphone accelerometers, gyroscope sensors, and inertial data from pre-loaded maps to identify parking spaces. PocketParker [35] detected users' movements and derived parking or unparking status using the accelerometer and GPS. ParkSense [36] detected WiFi beacons to determine if the driver had returned to the car or was in motion. UPDetector [28] employed various smartphone sensors, including accelerometer, Bluetooth, microphone, gyro, GPS, WiFi, parking payment apps, and user inputs, to detect drivers' behaviors. Villanueva et al. [48] introduced a vehicle detection method using the 3D compass of drivers' smartphones, enabling the detection of parking and adjacent parked vehicles. Krieg et al. [22] leveraged all available smartphone sensors to detect users' transportation modes, including car parking and unparking activities.

Crowdsourcing-based smart parking applications have been noted to introduce challenges, including concerns regarding the accuracy of information, user participation rates, and the presence of free riders, as demonstrated by Chen et al. [5]. Gupte and Younis [17] introduced a reputation-based approach to assess data reliability, where reputation scores increase each time a device submits non-corrupt data. TruCentive [18] is another crowdsourcing smart parking application that implemented a game-theoretical framework dynamically adjusting bonuses based on the proportion of honest players. Kifle et al. [47] proposed the UW-ParkAssist, a crowdsourcing smart parking application working in collaboration with UW-Police, which provides expert data to enhance the reliability of the collected information. Alternatively, opportunistic crowdsensing, as demonstrated by Coric and Gruteser [7], involves using ultrasonic sensors to collect street parking maps, similar to the approach taken by ParkNet [31]. iPark (Yang et al.) [50] created a parking map based on vehicular trajectories without real-time occupancy status information.

3.0 Empirical Study

As previously discussed, our key observation is the hearable difference between parked cars and empty parking spots. To understand this phenomenon and evaluate the feasibility of utilizing it for parking spot detection, we first conducted a empirical study. In our study, we mainly explored the following questions: *1. Where does the signal come from? 2. How can we use the signal for parking spot detection?*. In this section, we first present our experiment setup in our empirical study. Then we provide our findings and answers to the questions.

3.1 Experiment Setup

The experiment setup is shown in Figure 1, we put the detecting smartphone under the front window and used the camera on the Tesla vehicle to be the video ground truth. Aside from the video ground truth, we also implemented an ultrasonic sensor similar to the one used in Parknet [30]. By combining these two ground truth, we are able to gain a more precise knowledge of ground truth under all sorts of conditions. As Figure 1 illustrates, we explored 3 location of the smartphone to evaluate the impact of different location to the signal quality received. For our video ground truth, using the cameras equiped on Tesla Model 3, we are able to gain image from the left side, right side and rear side.

Our empirical study was conducted across 2 weeks with qualitative experiments and quantitative experiments. As Figure 2 illustrates, we covered an large area including campus area, park area and residential area.

We first explored the quality of received signal under ideal condition to figure out the feasibility of utilizing this sound of silence phenomenon for parking spot detection. Then we did a comprehensive exploration on all sorts of interferences on the received signal while driving, which we formed our knowledge of what kind of challenges we are going to face before we build a complete and functional system using this sound of silence phenomenon for roadside parking spot detection.

Figure 1: Empirical Study Experiment setup

We included different types of smartphones in the preliminary study that covers current major smartphone brands: Google Pixel2, Google Pixel4, Samsung Galaxy S22, Samsung Note10, iPhone 13pro and Huawei P20pro. As for the probing vehicle, we also included two cars that show differences in different aspects. The car we used are Tesla Model3 and Mercedes Benz GLA. These two automobiles not only stands for both electronic vehicle and traditional petrol vehicle but also represents sedan and suv. The video groudtruth is obtained using a web-camera, and the ultrasonic sensor was linked to our laptop through a Adafruit I2C board. Our preliminary study takes two weeks and includes about 510 parked cars with around 100 vacant parking spots.

3.2 Understanding the Phenomenon

3.2.1 The Source

The first thing we hope to figure our in our empirical study is the source of the hearable difference between a parked car and a vacant parking spot, which correspond to the first

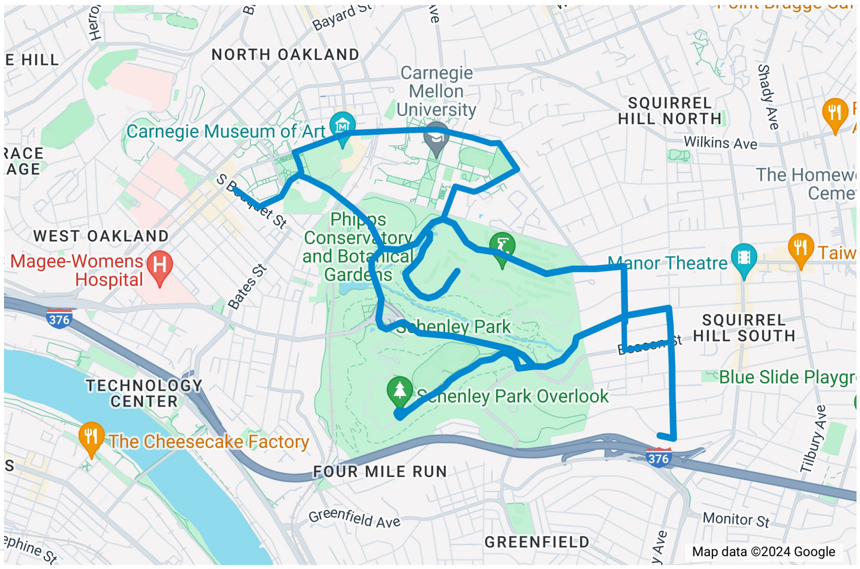

Figure 2: Empirical Study Experiment trace

question of *Where does the signal come from?* To understand this question, we performed Short Time Fourier Transform (STFT) on the signal received while driving pass the parked car on the roadside. Given noises under 500Hz is the dominating part in the received signal, we pass the signals through a high-pass filter of 500Hz to see the other existing components and received an STFT result as Figure 3 shows. In Figure 3, the period with a significant energy burst is the period when our probing vehicle is passing by a parked car on the road side. And the remaining areas represents empty spaces. Hence, in Figure 3, there are 3 cars in total that are collected in to this sample.

As Figure 3 illustrates, there are mainly 3 dominating components generated while the proving vehicle passes through a parked car other than signals with a frequency lower than 500Hz. The first frequency period is from 500-2kHz, which is in the frequency range of tyre noise according to [2]. The second dominating frequency period is from 8kHz to 9kHz, which is consistent with the frequency range of aerodynamic noises generated from the air passing by the moving vehicles according to [27]. The third frequency range that shows a pattern is around 17kHz, which is also generated from aerodynamic noises. As a results, we conclude the source of the signal from both aerodynamic noises and tyre noises.

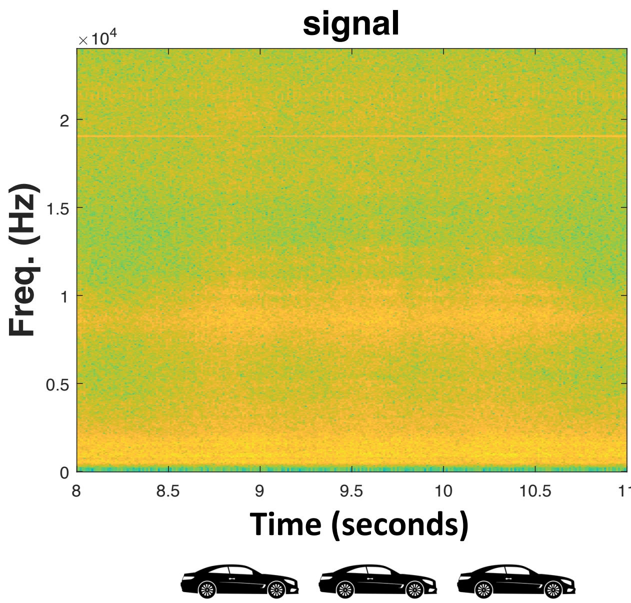

Figure 3: STFT Diagram of Received Signal

3.2.2 Affecting Variables

During the experiment process, we noticed certain variables that would affect the quality of the receiving signal. We further investigated into the affection of these variables, and gain more knowledge with the phenomenon.

1. Passenger Side Window

For starters, modern luxury cars (such as the Mercedes Benz we used in the empirical study) have good sound-proof technologies, which makes it relatively hard for the tyre noise and aerodynamic noise to penetrate through the windows into the interior spaces. As a result, opening the passenger side window is a precondition for quality data collection. We would like to highlight that although it is true that opening the window is necessary for quality data collection, it is not a critical issue for our system. Recall the idea of crowd-sensing, with enough probing vehicles in the city, there are always chances to have vehicles sending back audio data with good quality. In practice, we could set a credit score for each probing vehicle, which is based on the quality of data they sent back in the history. Using

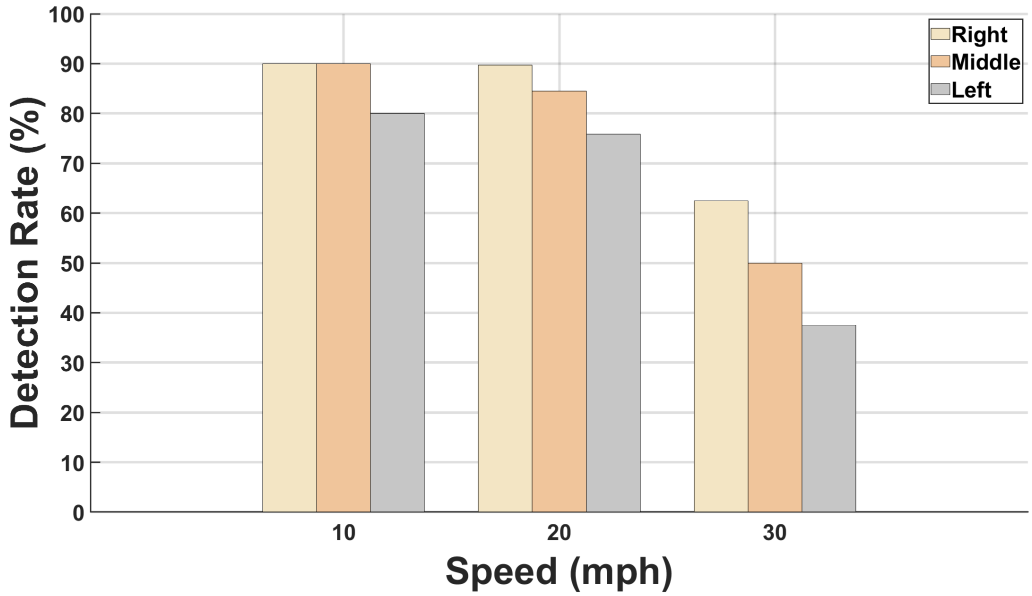

Figure 4: Detection Rate With Different Driving Speed and Smartphone Location

this credit score, we could prioritize the data that are highly likely to be of good quality, and use them first for prediction.

2. Driving Speed

Intuitively, the amplitude of tyre noise and aerodynamic noises are directly connected to the driving speed. With a higher speed, the tyre noise and aerodynamic noises will get stronger. However, this intuitive understanding is from a general perspective of all the noises considered. In order to understand the affection of driving speed to the specific noises ranging from 7.5kHz to 8.5kHz and 16.5kHz to 17.5kHz, we conducted a quantitative experiment to understand the affection of speed specifically.

As Figure 4 shows, the Y-axis represents detection rate, which is defined as the ratio between the number of parked car generated patterns that are actually recorded and the number of all recorded parked cars by the ground truth. To be more specific, the numerator of the equation represents the number of pattern generated by the parked cars that are recorded by the microphone. There might be cases where the patterns are covered by other

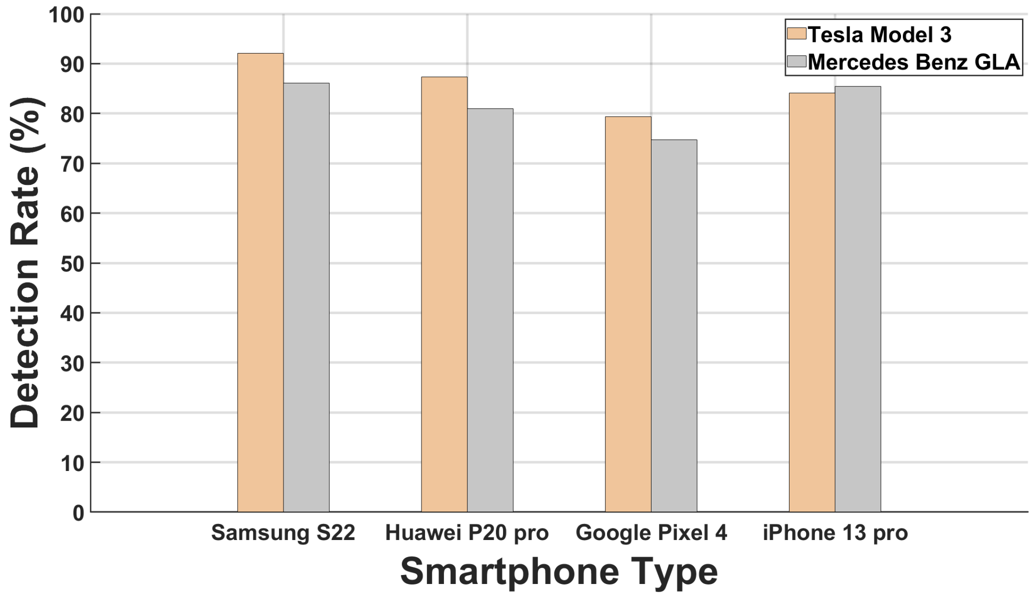

Figure 5: Detection Rate With Different Smartphone Type and Sensing Vehicle Type

noises and are not recorded clearly in our received data. We rule out the affected ones and only count the clearly recorded once. In doing so, we could understand how much the changing of condition variables will affect us in the detection process. The denominator on the other hand, is all the cars that were recorded in our experiment. We get this information from the ground truth, which is not affected by the condition variables. The higher the detection rate, the better quality the data shows for detection. We use this metric in the following quantitative experiment as well. Figure 4 illustrates the affection of speed and phone location. As Figure 4 shows, as speed increases, the detection rate decreases, meaning that the ambient noises get stronger as the driving speed increases, which makes it harder for us to detect the desired pattern generated by the parked cars.

3. Phone Location

The concept of phone location is defined as the location where we put the phone in the car, recall the experiment setup in Figure 1. As Figure 4 demonstrates, the detection rate decays as the phone location moves from the right to the left. This is because the signals transmit into the car interior through the passenger side window, which is on the right. As the signals transmit through the cabin, there is attenuation and also other ambient noises mixed with it, which all lead to the drop of signal quality. One thing delightful is that, although putting the phone in the middle (which is usually the place for phone stand) encounters shortcomings compare to putting the phone on the right, the gap between them is acceptable (smaller than 5% while the driving speed is lower than 20mph). Considering the normal speed limit for driveways with roadside parking as 25mph, putting the smartphone in the middle also shows the capability for detection.

4. Phone Type

As Figure 5 shows, we included 4 different types of smartphone in our empirical study, which are Samsung S22, Huawei P20 pro, Google Pixel 4, and iPhone 13 pro. The motivation of including different types of phone in the experiment is that different smartphone has different hardwares, which may lead to differences in the recording of the signals. To our delight, according to the results demonstrated in Figure 5, smartphone types does not affect the detection rate dramatically. The orange columns and the grey columns represent results collected from the same car using different smartphones respectively. The data in the orange columns have a range of 0.127 and a standard deviation of 0.0463 while the data in the grey columns have a range of 0.114 and a standard deviation of 0.0455. Both data has shown little variance, meaning that smartphone type does not strongly affect the detection rate.

5. Sensing Vehicle Type

In our empirical study, we have used 2 sensing vehicles: Tesla Model 3 and Mercedes Benz GLA. These 2 sensing vehicles are of good examples of electric and petrol-powered vehicles. We included these 2 cars in our experiment to (a) eliminate potential particularity of a single vehicle, and (b) see the difference between the electric and petrol-powered vehicles. As Figure 5 illustrates, the detection rate of Tesla Model 3 outperforms the detection rate of the Mercedes in general. Considering the fact that we have set the open window size to the same for both cars, the reason behind this result could be possibly attributed to the fact that petrol-powered vehicles have stronger engine noises than the electric vehicles. With the mixture of much more engine noises, the quality of received data from the Mercedes will be lower than the Tesla.

3.3 Choosing the Frequency Band to Use

3.3.1 Intuitive Selection

As discussed in the previous section, there are mainly 3 sources for the signal we receive while driving pass a parked car on the roadside, including tyre noises that ranges from 500Hz to 2kHz and aerodynamic noises around 8kHz and 17kHz. Consider the 3 frequency band, one major difference would be tyre noise lies in the audible band, meaning that they could be capture by human ears. If we are collecting data including tyre noise that ranges from 500Hz to 2kHz, we would also collect human speech that are associate to people's privacy. Imagining our system to be widely deployed, we will have our mobile system embeded into every driver's smartphone and collect audio data while they drive. The possibility of infringing people's privacy would prevent our system from promotion. As a result, the inaudible band signals is more preferred in our system design.

3.3.2 Quantitative Analysis

After we understand the source of the received signal, the following question is how we utilize these signals to realize our goal. To provide further knowledge to answer this question, we conducted a quantitative experiment using 2 cars in our lab to determine the accuracy of using these 3 bands for car length estimation. The car we used is a Tesla Model 3 and a Mercedes Benz GLA, with car length of 4.72 meters and 4.41 meters respectively. The speed in our quantitative experiment is controlled to 20mph while the probing vehicle is driving pass the parked car. The objective in our quantitative experiment is to control the parked car length and speed, such that we could have a understanding of using the 3 frequency bands to calculate the parked car length and therefore calculate the distance of a empty parking space. Considering using the Mercedes as the probing vehicle, and consider the speed to be 20mph, which equals to 8.94 meters per second, as long as we have the time duration of the signal generated by the parked car, we are able to calculate the distance of the parked car, which is now the Tesla model 3.

Figure 6 highlights the signal we received in the 3 frequency bands, Figure 6(a)-6(c) represents the time series data while Figure 6(d)-6(f) presents corresponding STFT diagram. As Figure 6(a) and 6(d) highlights, the signal in 500-2kHz band does not show clear pattern of the parked car. This is probably because the 500-2kHz band aligns with the frequency band with a variety of ambient noise, which makes the reflected tyre noise shadowed in other noises. From the results demonstrated in Figure 6(b) and Figure 6(e), we can see a clear pattern of parked car in the received signal. However, according to the calculation in Figure 6(b), the estimated car length of the parked car is $(1.013-0.344)*8.94 = 5.98$ meters, which is 1.26 meters longer than the real length of the parked Tesla Model 3. As for the 16.5kHz to 17.5kHz band signal, according to the results in Figure 6(c), the estimated car length is $(0.931 - 0.526) * 8.94 = 3.62$ meters, which is 1.1 meters shorter than the real length. One explaination for this phenomenon is that the 8kHz signal is generated before the probing vehicle and the parked car fully overlaps and will not disappear even if the Mercedes has already drove pass the Tesla. And for the 17kHz band signal, it requires the Mercedes to have a certain period of overlapping with the Tesla to generate the 17kHz band signal.

Given the above observation, the 500-2kHz signal not only contains strong ambient noises but also may cast privacy concerns on the data collection process. The 6.5kHz to 7.5kHz band signal and the 16.5kHz band to 17.5kHz band signal are both inaudible band signal, and they both show promising patterns of parked cars. However, the concern for the inaudible band signals is that they cannot measure the distance of the parked car correctly. Consider using the 6.5kHz to 7.5kHz band signal for parking spot detection, the estimated parked car length will be generally longer, leading to shorter estimation of potential parking spaces. As a result, using the 6.5kHz to 7.5kHz band signal alone may lead to false negatives in predicting available parking spots. When the same logic applies to the 16.5kHz to 17.5kHz band signal, potential false positive results is encountered.

Hence, in this thesis, we chose to leave out the 500-2kHz signal due to the lack of informative information and potential privacy concerns. We combined the 7.5kHz to 8.5kHz band signal with the 16.5kHz to 17.5kHz band signal, utilizing the information provided from both signals for vacant parking space prediction. We designed a DNN model to realize the prediction, based on the reasons explained in the following section.

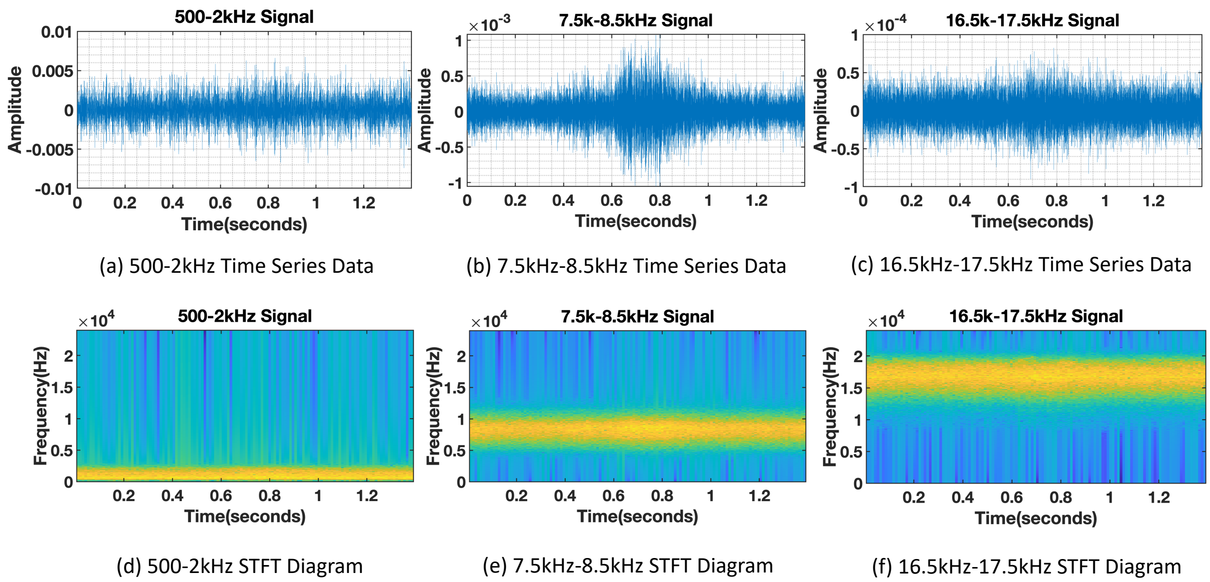

Figure 6: Quantitative Analysis of Received Signal

3.4 The design choice of using a model

In the following chapters, we designed a Deep Neural Network (DNN) model to make accurate prediction of the input audio. The reason we made this design choice rather than applying traditional signal processing approaches lies in the findings we have in our empirical study as well.

Through our empirical study, we discovered that there are a variant of ambient noises that would affect the received signal, some of which may even generate similar pattern to the ones generated by a parked car on the roadside. These patterns introduced by ambient noises may show similar energy or amplitude to the ones we hope to detect. More seriously, some ambient noises such as road bump or turning signals may overlap with the pattern of a parked car, leading changes to the pattern itself, which will suffer from misdetection if we use a straightforward hard-coded approach for parked car detection. Traditional signal processing techniques often requires pre-knowledge of the environment noise pattern. These approaches then apply existing techniques to eliminate the noise base on the defined pattern.

However, according to the knowledge we learned from our empirical study, not only does not the ambient noise have a clear pattern, but also the ambient noise are highly difficult to be separated from the pattern we want to capture. Hence, using straightforward signal processing techniques may be insufficient for accurate detection of the pattern generated from a parked car on the roadside.

On the contrary, data driven approaches could provide hyper knowledge about the input data that are often hard for human to detect. The underlying mystery features and inner connections of the data could possibly provide solution to our problem. Similar approaches have already been widely used in biomedical engineering, providing diagnosis in various scenarios, such as heartbeat and blood pressure measurements. Using Machine Learning techniques, we could design a model that suits our problem best. And with a self-created dataset that is diverse enough to include all possible scenarios, we can foresee a promising results.

3.5 Challenges

Through our empirical study, we have a clearer understanding of the issue and the design space we process with our observation. Meanwhile, we also have certain challenges to encounter to realize our system.

3.5.1 Automatic Segmentation

To begin with, the input to our system is a audio data stream collected from the probing vehicle. And clearly not all of the segments in that data stream consist useful information that indicates the parking availability information. Therefore, we need a segmentation module to break the gap between a data stream to segments that contains useful information. Considering deploying our system to mobile devices and the consistent data collection, we need an automatic segmentation algorithm for our system, which will do its best to rule out undesired segments and preserve the informative periods in the collected data.

3.5.2 Parked Car Detection

Considering detecting a parking spot, the first step is to determine the parked cars, especially in the road segments that do not have a clear parking mark on the roadside. To detect the parked cars, the nature behind it is to distinguish the pattern of a parked car with all other patterns. As discussed in the previous sections, using a signal processing approach is not enough for the goal here due to the indistinguishability between ambient noise and pattern generated by the parked cars. Also, using neither the 8kHz band signal nor the 17kHz band signal alone is sufficient to provide accurate estimation of the parking spot. Consequently, we need a solution to (a) combine the information from both the 8kHz band signal and the 17kHz band signal and (b) distinguish the pattern generated from a parked car from vacant spaces and other ambient noises.

3.5.3 Formalizing Parking Availability Information

After we get the information regarding the parked cars on the roadside, there is still a gap between this kind of information to the parking availability information. To bridge this gap, we need to calculate the distance between two consequent parked cars, or in other words calculate the gap between two adjacent parked cars, and check out if this distance is large enough for a normal car to park. To realize this functionality, we need to factor in the speed of the probing vehicle and also consider the normal length of a vehicle.

4.0 Design

In this section, we first provide a high-level overview of our whole system. Then we present details regarding each module.

4.1 System Overview

In this thesis, our system mainly consists 3 modules: (1) a pre-processing module for data automatic segmentation, (2) a Deep Neural Network (DNN) model for empty parking space prediction, and (3) a post-processing module to refine the results and finalize the output of our system.

Considering the input to our system as audio data stream, the pre-processing module is used to automatically segment the data stream into segments of same length, which will later input to the DNN model for empty parking space prediction. Also, the segmentation process should also kick out segments that (a) do not contain information, or (b) contains information that are clearly not what we are looking for.

As discussed in the previous chapters, the purpose of our DNN model is to make empty space prediction while handling interferences from ambient noises. The input to our model is the audio segments from the previous pre-processing module, while the output of our model is a sequence of probability showcasing the likelyhood of the corresponding data point to be a parked car or a empty parking space.

After we have the output of the DNN model, we have a prediction of parked cars in the input period of data stream. To finalize the parking availability information, we need to calculate the distance between two adjacent parked cars, and determine if the vacant space is enough for a normal car to park. By counting all the predicted available parking space, we have the estimation of parking availability information to the corresponding road segment, and could use it to update the database and front-end displaying results to the user.

4.2 Pre-Processing

In real-world deployment, our probing vehicle will continuously collect signal as they drive through the areas. Hence, the input to our system is in the form of an audio data stream. As a result, before delivering the data to the model for prediction, we need to segment the data into period that (a) includes desired information and (b) satisfy the constraints for our model, such as the input audio length, etc. Segmenting audio periods into certain length is easy and straightforward. However, choosing the clean audio periods that contains information is the tricky part. In our pre-processing module, we mainly consider distinguishing the desired segments with the segments that suffer from external interferences as defined in 3.3.3 and the vacant periods that has no cars on the right.

To realize automatic segmentation as required, we propose a signal processing pipeline as illustrated in Figure 7. As Figure 7 shows, after the probing vehicle collects audio data, 2 band pass filters are first applied. The frequency band applied are 7.5kHz to 8.5 kHz and 16.5kHz to 17.5kHz, as we discovered in our empirical study. Then, considering strong ambient noises may have a significant amplitude compared to the pattern we desire, we introduced a time domain amplitude threshold to our pipeline. In doing so, we are able to detect the data point with significantly high amplitudes, which are noises and may affect successful segmentation. After we detect the significant noises, we apply a scaling factor to reduce the amplitudes of these data points. The time domain amplitude threshold and scaling factor are both empirically set, in our final implementation, the threshold is set to 0.001 and the scaling factor is set to 0.01. One thing worth noting is that we did not scale down the significant noises to the same amplitude to our desired pattern, instead we scale these data points to an amplitude that is slightly higher then the desired pattern's amplitude. This choice is made to prevent the possibility of introducing false positive results, namely introducing patterns that do not exist.

After scaling down the significant noises, we normalize all data points to increase the amplitude of each data point without changing the relative amplitude of each data point. Then we perform segmentation using a energy based threshold. We used a open-source tool called auditok on Github, which is a segmentation algorithm using a energy based

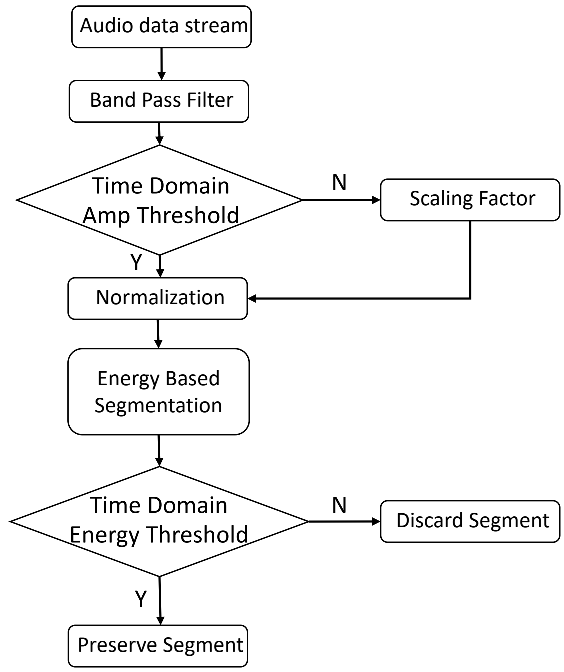

Figure 7: Signal Processing Pipeline

threshold on energy calculated on each sliding window. We set our segments to 3 seconds each, considering to contain enough data points for a parked car and also having fixed length of data sample will facilitate the training process of our following model.

In the open-source tool, a signal lower bound threshold is applied to segment out segments with audio activities. This result in segmenting out segments including ambient noises as well. In order to further kick out undesired segments, we introduced another time domain energy threshold including a upper bound and a lower bound. The threshold are set empirically, which includes the energy period of a normal pattern we desire. After this threshold, we get the segments that are highly likely to be the pattern generated by a parked car, which come from the reflected tyre noise and aerodynamic noise between the probing vehicle and parked car. And it is certain that in our segments, there are still segments that (a) do not include the pattern of a parked car, (b) include ambient noises that have similar energy with the pattern of a parked car. We then pass all the segments from our pre-processing module into our model for prediction. Through our training, our model is able to handle these negative samples and make accurate prediction to an acceptable level of precision.

4.3 Deep Neural Network Model

4.3.1 Model Architecture

After the aforementioned pre-processing module, we have the segments from the audio data stream collected by the proving vehicles. These segments contains segments of desired patterns of parked cars, and also the segments that may include ambient noises. Our model is designed to distinguish parked cars, ambient noises and vacant spaces. Given this consideration, we formulate this task into a binary classification problem, in which 1 represent empty space and 0 represent a parked car on the roadside. In our training process, we include enough sample of ambient noises to help our model capture the differences between these noises and the pattern generated from a parked car. More details regarding the training process will be discussed in the following chatpters.

The segments we have from the pre-processing modules are time series audio signals of 3 seconds duration. As Figure 8 illustrates, we first apply Short Time Fourier Transform (STFT) to these segments to transform them into frequency domain. The decision of applying STFT is to include one more dimension of information from the frequency domain, which now will provide us with information from amplitude, frequency and time. The signals we preserve are in the frequency of 7.5kHz to 8.5kHz and 16.5kHz to 17.5kHz, which are both necessary for our model. As discussed in our empirical study, using neither of these 2 signals are sufficient enough for accurate car length estimation, leading to possible false positive and false negative. By incorporating the information from both band, we will get more promising results. After the STFT, we perform channel width concatenation on the 2 signals, to form a single sample with channel from both input sources, which includes information from both frequency bands.

Our model is a combination of multiple Convolutional Neural Network(CNN) blocks followed by 2 Fully Connected Layer. Given the input as STFT diagrams of the 2 audio sources, we chose to use a 2 dimensional convolutional neural network block for feature extraction. The activation function we use is ReLU, which is a standard choice and works well for our scenario. We added batch normalization and dropout operations to our model

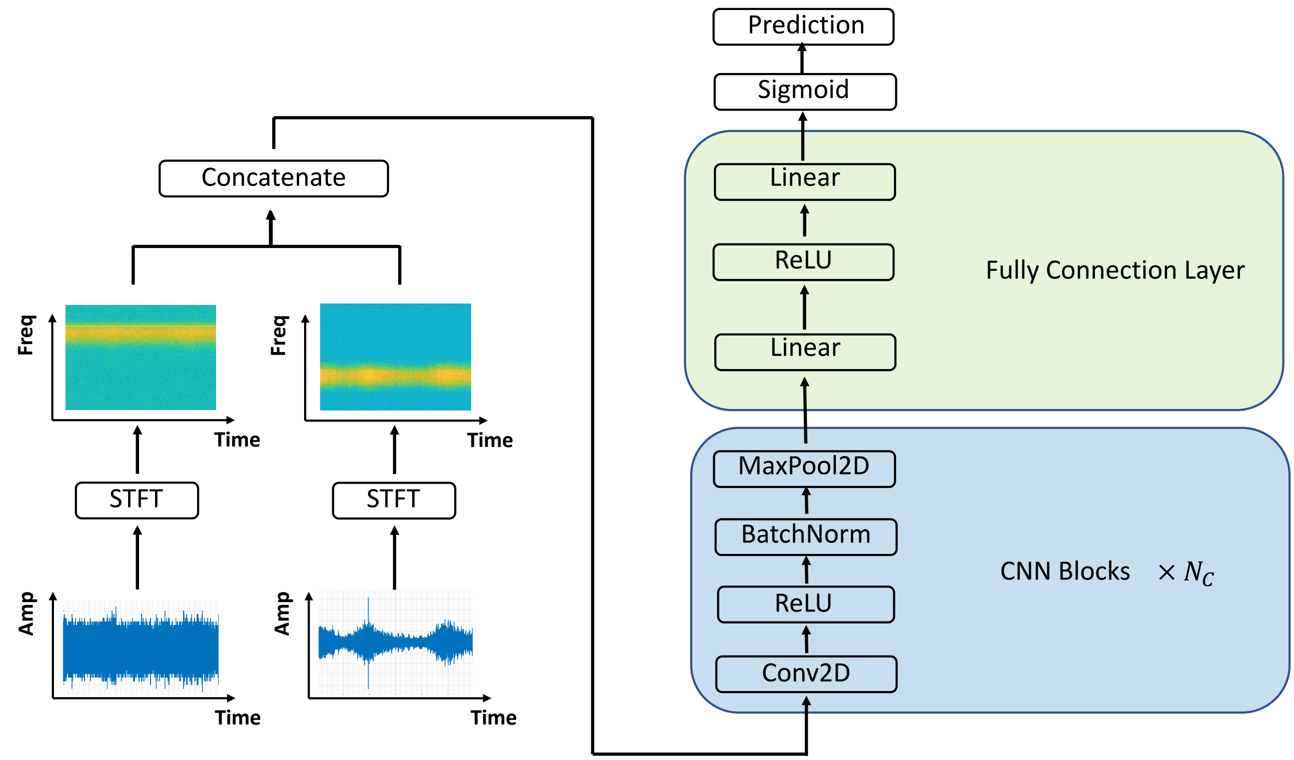

Figure 8: Model Architecture

to prevent the model from over-fitting. We included padding operations to preserve the boundary information, and included a pooling layer to focus on the most significant features. After the consequent CNN blocks, we added 2 fully connected layer with the ReLU function to further extract and combine extracted features, which finalizes the learning results of our model. The last block of our model is a sigmoid unit, we apply this unit to guide our model to perform 0 and 1 predictions. In other words, we use the sigmoid unit to guarantee the output will be distributed around 0 and 1.

4.3.2 Training Strategy

During our first attempt of training, we discovered that the model is more capable of making the right prediction with the middle part of the input sample and shows insufficient ability in making right predictions with the boundary part of the input sample. This is probably because the middle part has both the information from its previous periods and the information from its following periods. On the contrary, the boundary parts will lack

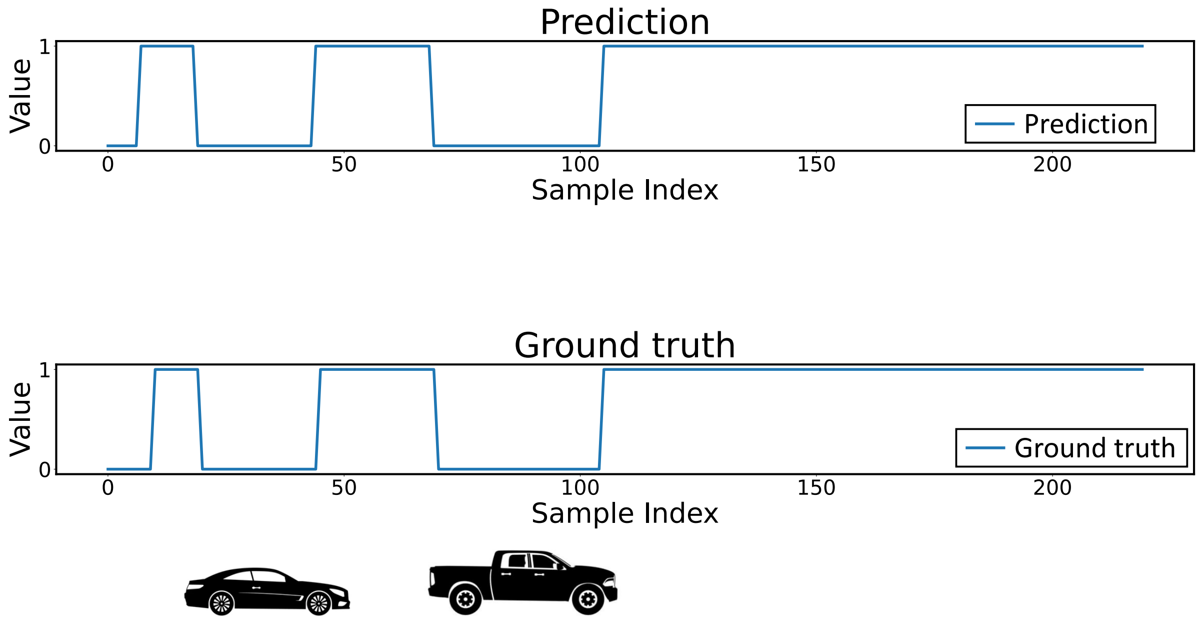

Figure 9: Model Output

either the information from its previous period or the information from its following period. Therefore, a straightforward training process may be unsuitable for our scenario.

Given the observation of the model being more capable of extracting features in the middle part of a sample, we made a change to our training process to only focus on the middle part of each sample. To be more specific, we have 90 data points in each sample, and we chose to use only the 30 data points in the middle for the training, which are data point 30 to 60. In doing so, we could guide the model to focus on the part which will provide the most accurate information. One thing worth noting is that by selecting only the middle part for training, we will break the consistency of all the sample, meaning the beginning and ending part of each sample is missing from a general perspective. To mitigate this issue, we performed overlapping to our dataset with a corresponding overlapping factor of 1/3, which is consistent with the selection to each data sample. Consequently, every 3 samples will form the entire original sample, and we can still get a complete picture of the whole input data stream while focusing on the most informative part of each sample. Through our experience, this training strategy greatly improved our accuracy.

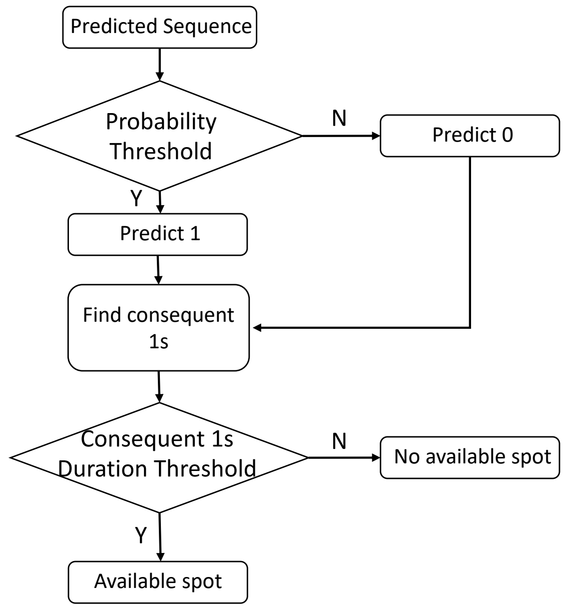

Figure 10: Post Processing Pipeline

4.4 Post-Processing

The output of our model is a sequence of float type numbers in the range of 0 to 1, which implies the possibility of the data point to be a empty space for parking. This information is not equal to parking spot information considering that a empty space have to be long enough for a car to park, especially in areas without parking marks on the ground. Hence, to refine our results, we applied a post-processing module for the formal results of parking availability information.

As Figure 10 shows, with the sequence we obtained from our model, we first apply a hard-coded threshold to finalize the predictions to 0s and 1s. Our approach is to use a probability threshold of 0.5 to label data points less than 0.5 to 0 and data points higher than 0.5 to 1. This is reasonable considering the output of our model is highly deviated from 0.5, the output data points are often distributed around 0.9 and 0.1. As a result, using a straightforward hard-coded threshold is suffice for our purpose of finalizing the predictions. In order to find empty spaces that are large enough for a parking spot, we need to find the consequent 1s

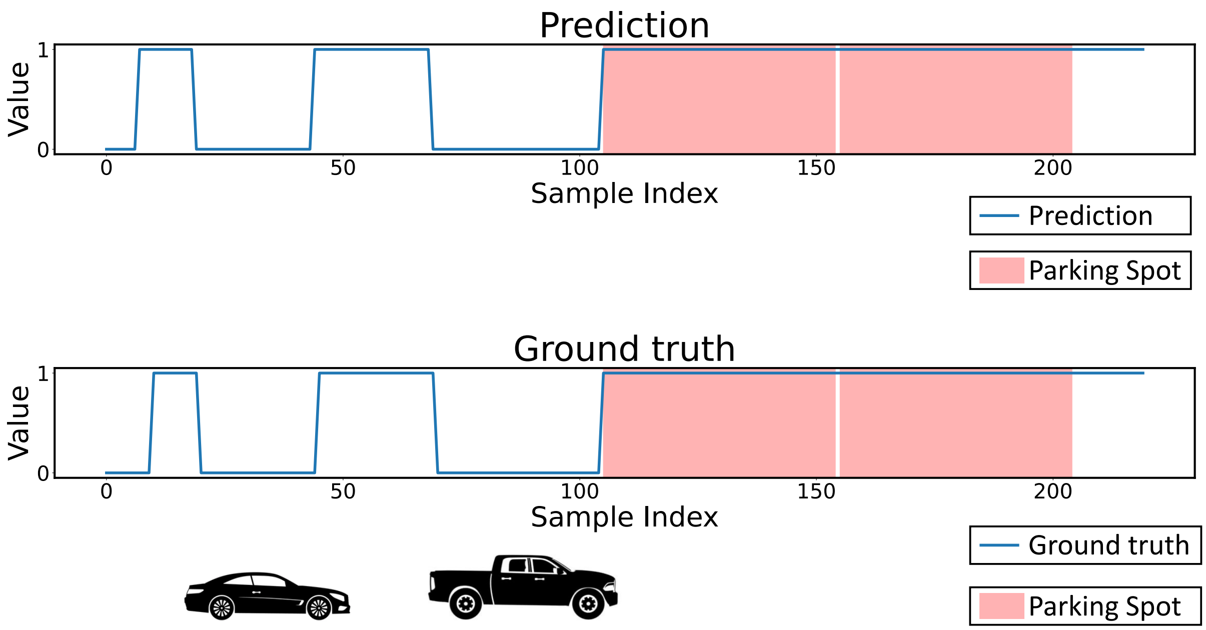

Figure 11: Post Processing Output

in our outputs. To do so, we first connect all the samples in our evaluation phase, which formulates the stream of data collected from a certain period. Then, considering the normal length of a parking spot to be 6 meters, and the speed of vehicle in our data collection is 20mph, a parking spot would last for 0.671 seconds. Hence, a parking spot in our dataset would be consequent 1s that last for more than 0.671 seconds. In order to realize detection, we first find all the consequent 1s in the stream we formulated from our evaluation dataset. Then we apply a duration threshold of 0.671 seconds on all the consequent 1s periods, and filter out all the periods that are predicted as an available parking spot.

Figure 11 presents a illustrations of the final results of our system. The blue line represent the prediction of our system while the blue line in the lower figure is the corresponding ground truth. The red rectangles depicts the detected available parking spots. As the results in Figure 11 demonstrates, our system shows a great ability in finding availability parking spots on our evaluation dataset, both in the quantity and location.

5.0 Evaluation

In this chapter, we first present details of our dataset, including the experiment we conducted to create the dataset and the information of the dataset. Then we define the metrics we focus on in this thesis to evaluate the performance of our system. We decide our concentration on certain metrics through a simple user study we performed among the members in our lab. Finally, we present our results on the dataset we have and present our findings and observations for future work.

5.1 Experiment

As Figure 12 illustrates, in our final experiment which we collect data to create our own dataset, we used two smartphone to collect data from both the phone stand location and the location close to the ground truth. We used a RGBD camera to capture both the video and the distance information. We wrote a Python script to focus on the distance value of a single pixel in the frame, and record all the distance value in a CSV file as output.

The smartphone location are selected consider the following factors: (1) The common position of a smartphone is the phone stand inside the car, (2) In order to align with the ground truth, we need to put a smartphone right next to the ground truth. As Figure 12 indicates, we collect video together with the distance data from the RGBD camera, the distance data are inherently collected from the ground truth given that they measure the distance from the probing vehicle to the far most distance the RGBD camera could measure.

In practice, the distance value we retrived from the selected pixel shows the following features: (1) When there is no parked car on the roadside, which represent a empty parking space, the distance value is around 5 meters. (2) When there is a parked car on the roadside, the distance value is around 2 meters, or in some cases, due to the reflection from the car surface, we may receive unexpected 0s. Given the above fact, we could simply set a threshold of 3 meters to encode the distance values we retrived from experiment. After this hard-coded

Figure 12: Evaluation Experiment Setup

encoding, we would have the desired data in which 0 represents a parked car and 1 stands for a vacant parking space. As mentioned above, the ground truth value are automatically recorded by our Python script and output to a CSV file.

One thing worth noting is that we may encounter unexpected 0s in the collected data due to the strong reflection of day light from surfaces like the car surface. For these 0s, we manually adjust these values to the correct value according to the video ground truth we simultaneously collected with the distance value. This is a time consuming yet necessary process since the ground truth is the source of the knowledge for our model, the correct ground truth will guarantee the good performance of the training process.

In this thesis, we have several assumptions made, which reflects in our collection of data. To begin with, with the knowledge we obtained from our empirical study, we noticed that opening the passenger window guarantees good signal quality. Base on this observation, we set the open size of passenger side window to 16cm and remains this setting during the collection of our data. Secondly, driving speed affects the received signal as well, and it also affects the calculation of parking space in the post-processing module. To simplify the scenario in this thesis, we controlled the driving speed to 20mph during the experiments.

Finally, in this thesis, we only considered the inevitable interferences, including scenarios such as driving through road bumps, encountering wind while driving, noises generated from driver's movements, etc. Interferences such as the surrounding moving vehicles, music played inside the car and human talking inside the car are not considered in this thesis and included in future directions and future work.

Our experiment for dataset creation is conducted through a period of 2 weeks, including data from 10 hours of audio and video collection. The dataset took us another week to build, with 2512 samples in total. The samples contains a variant of scenarios, including (a) pattern generated by parked cars without interferences, (b) negative samples in which there is no object on the roadside, which means the sample is a empty parking space, (c) samples with interfenrences from ambient noises, these samples may be of patterns generated by parked cars or simply empty parking spaces. As previously discussed, the objective for our model is to make precise binary prediction of parked car or empty parking space while tolerating the presence of ambient noises. We include data of all kinds considered in our model objective to provide reletive knowledge to our model.

After the whole dataset is built, we separated the dataset into training set and testing set randomly by a factor of 0.8 : 0.2, resulting in a training set of 2009 samples and a testing set of 503 samples. We used the API provided by Python scikit-learn package for dataset split and randomly assigned data samples to both set to guarantee fairness to the evaluation process. During the creation of the whole dataset, we also built two partial dataset base on different weather condition. To be more specific, the first dataset contains 917 samples with moderate weather. The second dataset contains 1435 samples with 518 additional samples of bad weather condition, these data samples are collected while the weather is windy and large wind affects the audio data collected. Finally, the whole dataset contains all samples collected, with 1077 samples of both good weather condition and bad weather condition data samples.

5.2 Metrics

To evaluate our results, certain metrics are applied for the evaluation process. Aside from all the standard metrics to evaluate the results of a DNN model, we also focused on certain metrics base on the consideration of real-life scenario of our problem. We conducted a micro-user-study to understand what people truly value for a parking availability information gathering problem and focus on the significant metric during our evaluation.

5.2.1 Overall Metrics

Considering a standard DNN model, the metrics to evaluate its performance are as follows:

1. **Accuracy**: The ratio of correctly predicted observations to the total observations.

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Observations}}$$

2. **True Positive (TP)**: The correctly identified positive cases.
3. **False Positive (FP)**: The cases that are incorrectly identified as positive when they are actually negative.
4. **True Negative (TN)**: The correctly identified negative cases.
5. **False Negative (FN)**: The cases that are incorrectly identified as negative when they are actually positive.
6. **Precision**: The ratio of true positive cases to all the cases that are predicted as positive.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

7. **Recall**: The ratio of true positive cases to all the cases that are actually positive.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

8. **F1 Score**: A measure of a test's accuracy that considers both the precision and the recall.

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

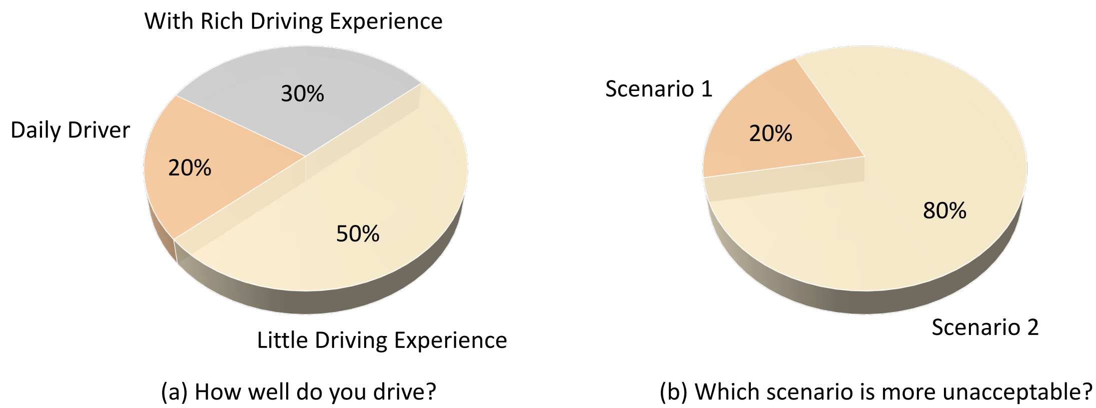

Figure 13: Micro User Study

In our evaluation, we look into all the metrics above and specifically focus on the True Positive Rate (TPR), False Positive Rate (FPR), and Precision score base on the following micro-user-study.

5.2.2 Micro User Study

In order to better understand the user need for parking availability information, we conducted a micro user study inside our lab. Considering in this thesis, we are not focusing on the real dissemination of information, we do not concentrate on what kind of information the user wish to see. Alternatively, we wish to gain knowledge about how we should finetune our system, especially our model to reach a performance that suits user needs best. Hence, in our micro user study, the major question we ask is that *Which scenario would you find less tolerable?* We provide 2 option to this question and ask our user to make a choice from them: *(1) The system tells me there is no available parking spot, but there actually are available parking spots. (2) The system tells me there is a available parking spot, but there is actually none when I drove there.* The first scenario correspond to the system making False Negative responses, misguiding the user to rule out the current street for parking. The second scenario represents the case when the system makes False Positive responses, which mislead the user to the street but only to find there is no place for him or her to park.

1. Participants

As mentioned above, our micro user study is conducted among the members in our lab, which include 10 person in total. The user characteristic we mostly care is their understanding of the parking problem. Hence, before we ask them to make a choice from the two scenarios, we first ask them about their driving ability. As Figure 13(a) illustrates, among the 10 participants, 2 of them are daily drivers, 3 of them have rich driving experiences but do not drive daily, and the rest 5 of them do not classify themselves as good drivers.

2. User Responses

Figure 13(b) shows the answer we received from the participants. The majority of the participants (8 out of 10) considers the second scenario more unacceptable, which is the system making False Positive predictions. The reason they generally give is that this may cause them effort and time to drive there only to find they have to search for a parking spot again, which is a waste of time.

There are also 2 participants who selected the first scenario to be more unacceptable, which is the system making False Negative predictions. They give their reason as they would consider the system to be unhelpful in this case. If all the street they are looking at seems to have no parking spot available, they would have to get back to searching blindly by themselves.

Although all the metrics are important for our evaluation, base on the majority preference, our system should make the best effort to avoid False Positive and try to maintain as much True Positive results as possible.

5.3 Results

In this section, we first present the overall performance of our system using the whole dataset we created for evaluation. Then we consider the different environment condition for the data and compare the performance of our system under different conditions.

5.3.1 Overall Performance

As mentioned in the previous sections, we created 3 dataset containing data under different weather conditions. We trained our model on all 3 datasets and evaluated their performances. Table 1 presents the details regarding the 3 datasets we used in our evaluation. One thing worth noting is that the 3 datasets are iteratively created, meaning dataset 2 contains all the samples in dataset 1 and dataset 3 contains all the samples in dataset 2. As for the weather condition, the most concerning weather condition in our scenario is the wind, since the sound of wind will affect the signals received by our smartphone microphone. Hence, the weather condition we discuss here mainly focus on the wind.

Table 1: Datasets Description

Dataset ID	Size	Weather Condition
1	917	Moderate
2	1435	Windy
3	2512	Good

As Table 1 illustrates, the size of the 3 dataset are 917, 1435 and 2512. The first dataset contains data sample collected from moderate weather conditions, meaning there is no heavy wind encountered. In dataset 2, the 518 data samples added to dataset 1 includes data collected from 2 days with heavy winds. And in dataset 3, the added 1077 samples are data collected from a day with almost no wind, providing us with some good quality data.

Table 2: Performance Over Different Datasets

Dataset ID	Accuracy	TPR	FPR	TNR	FNR	Precision	Recall	F1
1	88.10%	97.93%	35.73%	64.27%	2.07%	0.8692	0.9793	0.9209
2	92.00%	95.60%	20.97%	79.03%	4.40%	0.9428	0.9560	0.9493
3	93.51%	96.55%	19.78%	80.22%	3.45%	0.9553	0.3655	0.9604

Table 2 presents the performance of 3 used dataset among all the metrics considered in the evaluation. One thing worth noting is that in the evaluation, the positive value (1)

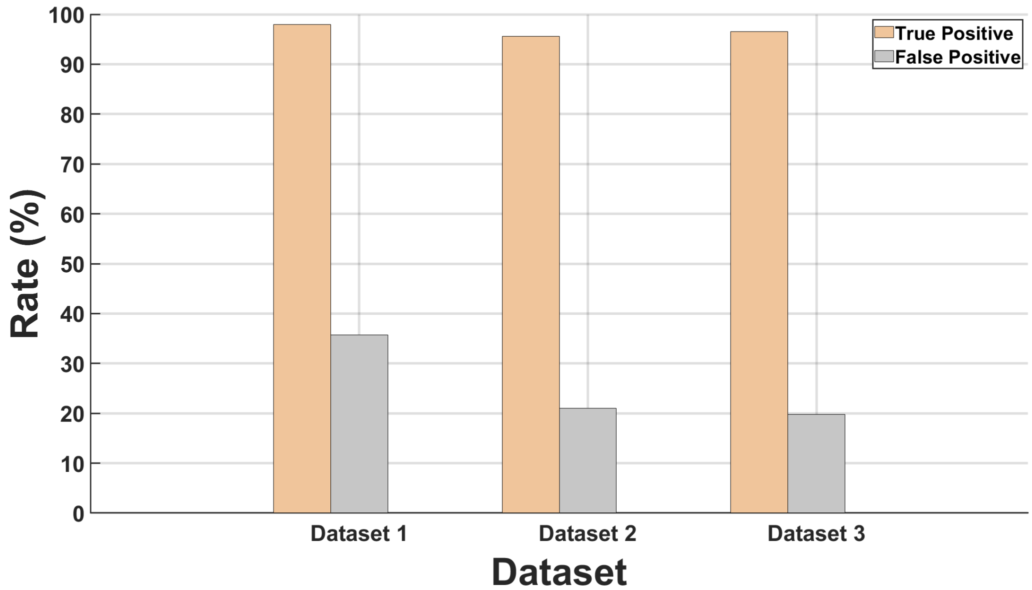

Figure 14: True Positive Rate and False Positive Rate

represents empty parking space while the negative value (0) represents parked cars. As Table 2 illustrates, our model could reach an accuracy of 90% in predicting empty parking spaces, with the best case accuracy of 93.51%. For empty parking space prediction, our model could achieve an at least 95% accuracy in making the right prediction. And as the number of samples increases, our model makes fewer false positive predictions about empty parking spaces. For parked cars prediction, as the number of samples increases, our model shows higher capacity in making the right prediction about the parked cars on the roadside. Meanwhile, to our delight, the False Negative Rate remains relatively small all the time, meaning that ambient noises does not easily affect our model, and our model shows the ability to distinguish the pattern generated from parked cars between the patterns generated from ambient noises.

5.3.2 Performance Under Different Condition

Base on the aforementioned micro user study, the goal of our model is to achieve True Positive Rate (TPR) as high as possible while maintaining the False Positive Rate (FPR) as low as possible. By looking into the performance of different datasets, we gain a deeper understanding of our model and the affecting variables.

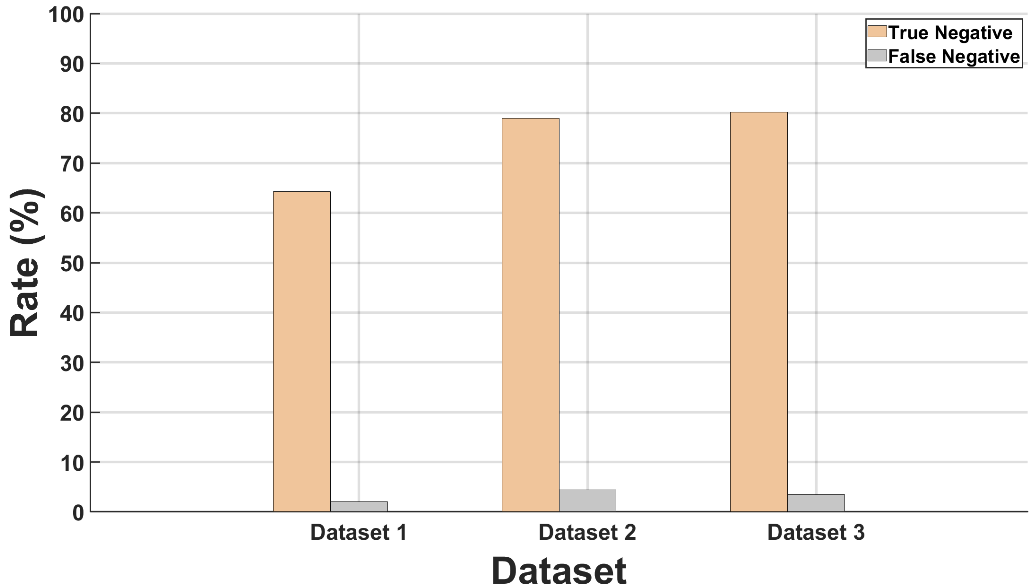

Figure 15: True Negative Rate and False Negative Rate

1. The impact of dataset size

According to the results demonstrated in Table 2, with the increase of dataset size, the accuracy increases as well. This showcases the feasibility of the initial design principle of utilizing data driven approach to provide understanding of the underlying features. With our final dataset (dataset 3), our model could achieve an accuracy of 93.51% in predicting vacant parking spaces, which is delightful and promising. Also, with dataset 1, which is the smallest dataset, we could also achieve an accuracy of 88.10%. This result is also quite acceptable, which further enhanced our confidence in the data driven approach to solve this problem.

2. The impact of weather condition

As previously discussed, the weather condition we consider is mainly the wind due to the fact that wind will strongly affect the quality of received signal. Table 1 summarizes the weather condition of the three dataset we used in our evaluation. Dataset 1 contains only data samples collected under moderated weather condition while the other two datasets includes

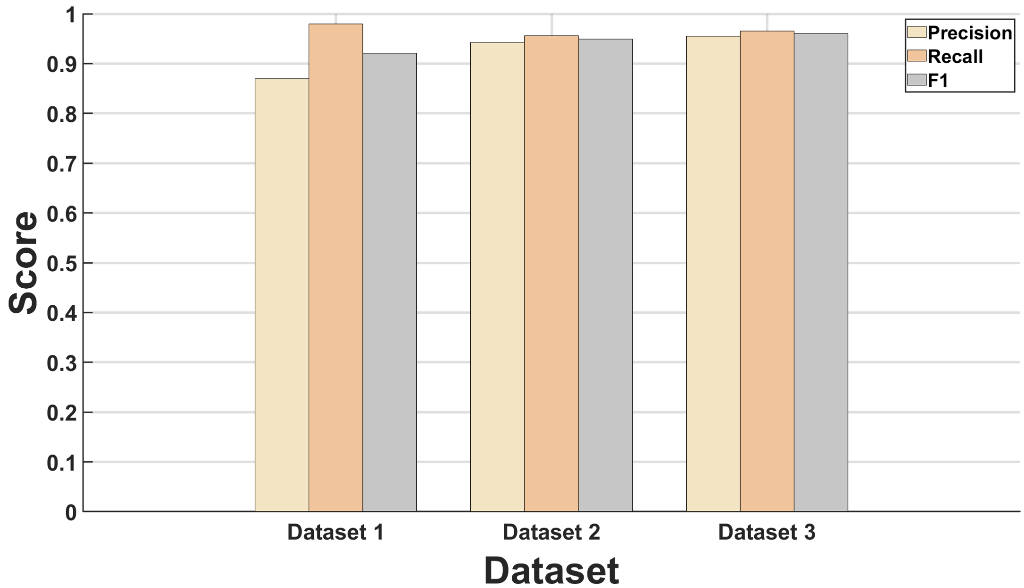

Figure 16: Precision, Recall and F1

data samples collected under heavy wind weather condition. Dataset 3 has additional data samples collected on good weather conditions with merely no wind encountered.

Base on the result in our micro user study, the goal of our system is to achieve high True Positive Rate while maintaining a low False Positive Rate. Figure 14 illustrates the performance of our model over the two metrics on different datasets. As Figure 14 shows, the model trained on dataset 2 and dataset 3 has lower True Positive Rate comparing to the model trained on dataset 1. And the TPR of dataset 3 is higher than the TPR of dataset 2. This is reasonable considering the wind will introduce more ambient noises and also affect the pattern generated by the parked cars, which resulted in the fact that dataset 2 and dataset 3 shows lower TPR. And if we consider the False Negative Rate(FNR) in Figure 15 together, the FNR of dataset 2 and dataset 3 is higher than dataset 1. This means that our model makes more prediction of parked cars (0s) using dataset 2 and 3, which aligns with our initial observation that wind will introduce ambient noises that could potentially lead to FNR of misclassification of parked cars.

Figure 14 also demonstrates the FPR on all three datasets, showcasing a continuous decrease of FPR as the dataset size increases. Meanwhile, the TNR demonstrated in Figure 15 shows a steady increase trend as the dataset size increases. This phenomenon is attributed to the increase of the number of data samples. With the increase of dataset size, more scenarios are presented to the model, which provided the model with more knowledge to extract features that best represent the pattern we are looking for. The increase of TNR highlights the increase in the model's ability in capturing the pattern generated by the parked cars, even though data collected under heavy wind weather condition introduced more interferences of ambient noises.

3. The trade-off between dataset size and weather condition

In our evaluation, the major variables are the dataset size and the weather condition when we collect the data. With the above observations, there exists a trade-off between these 2 variables that affects the performance of our model. For instance, the steady increase in TNR showcases that although data samples with ambient noises are introduced in the later two datasets, which may cause distractions in detecting parked cars, the increase of dataset size still guarantees accurately deteting the parked cars on the roadside.

Figure 16 provides the results regarding precision, recall and f1 score, which will provide us with a more general understanding of the model's ability to capture the positive data points. From a general perspective, both three metrics are relatively high, showcasing an acceptable ability of our model to capture the positive class, which represents vacant parking space. As Figure 16 illustrates, the precision score shows a continuous increase as the dataset size increases while the recall score encounters a drop when the data collected under heavy wind weather condition is introduced into the datasets. This phenomenon demonstrate that with the increase of dataset size, the reliability of our model's prediction increases. On the other hand, heavy wind weather condition will trick our model to detect less true positive data points, which aligns with our previous analysis that heavy wind will introduce ambient noises that may be classified as the pattern of parked cars.To our delight, the f1 score in Figure 16 provide us with a more comprehensive understanding that in general, our model's ability of capturing positive samples increases, meaning that our model is better at detecting the vacant parking spots.

5.3.3 Findings and Takeaways

According to the above evaluation and analysis, the first thing we learn is the feasibility of our approach. Our results has demonstrated promising perspective in leveraging the inherently generated signals from moving vehicles for available parking spot detection. The data driven approach which we conducted in this thesis is also proved to be effective. Despite the possibility of low quality data leading to misdetections, our results has demonstrated the capacity of data driven approach as long as the dataset is diverse enough that includes enough scenarios.

Aside from the promising results we have seen in our evaluation, we also identified several important variables that would affect our model. Bad weather condition is one major interference we have seen in our evaluation, which could trick our model to making false negative predictions that considers ambient noises to be potentially a parked car. This prompt us to thinking future directions to handle all kinds of interferences. As mentioned in the previous chapters, our work in this thesis has made certain assumptions to rule our some of the interferences that we may encounter in real-life scenarios. The results we seen with the interference of wind has further proved our initial understanding about the necessity to handle possible interferences.

6.0 Future Work

In this thesis, we built a end-to-end system that verifies the feasibility of leveraging the inherently generated signals from a moving vehicle to perform parking spot detection. Although the results are promising and provide confidence for such an idea, there are several assumptions that lie underneath our system, and there are certainly spaces for these assumptions to be removed.

To begin with, one major assumption in this work is that there are no interferences from the surrounding vehicles, both in the opposite direction and the same direction. Consider a one-way road with roadside parking, this may not be a concerning issue for our current system. However, for a two-way road with cars moving on the opposite direction, there may be signals generated by other moving car to the left of our probing vehicle. Such signals will also be recorded and may lead to false positive results if we do not rule out these certain signals. Moreover, consider a driveway with multiple lanes in the same direction, there are also moving cars in the same direction. And it would not always be the case that our probing vehicle is driving in the lane that is far-right. In these cases, the probing vehicle may record signals generated from the cars moving in the same direction, which will lead to potential false positive as well.

Another major assumption in this thesis is that the speed of the probing vehicle is 20mph in our evaluation. Although it is fair to make this assumption since the common speed limit for a street with roadside parking spots is around 20mph, it is nonetheless not always the case. To further adjust to real-life scenarios, one possible direction is to eliminate the assumption for speed, and take into account the real speed of the moving car. This would require additional access authentication to the smartphone OS for GPS authority to calculate the speed. Also, the speed needs to be as real-time as possible given that a slightly change to the speed may result in mis-classification. Consider standard speed calculation that are based on the offset of location divided by the time duration, maintaining real-time speed information is a potential challenge to solve.

Finally, as mentioned above, we have a fixed opening passenger window size of 16cm in our final experiments. Considering the fact that modern vehicles have certain technologies to reduce the noises inside the cabin, closing the passenger side window could potentially harm the performance of our system. Since opening of the passenger window is uncontrollable, one possible solution to this problem is to leverage the power of crowd-sensing. With enough cruising vehicles in the area, it is possible that a cruising vehicle that collects high quality signals will pass through a certain period. By adding another module to check the signal quality in the beginning of our system, we can detect the vehicle capable of collecting high quality signals, and give more trust to the results transmitted back from these vehicles. Realizing this idea and investigate the boundary for satisfying results is definitely another direction to look into in the future.

To remove these assumptions and push further to the real-life scenarios, a more sophisticated model may be the solution. With the increase of computational power, computer vision has been a rapidly developing area, leading to multiple extraordinary breakthroughs. Recently, there has been a trend for multi-modal fusion in designing DNN models. More specifically, people are leveraging the information from both the audio input and the video input to guide the model to learn. This additional information from the video will provide knowledge that the ground truth we have in this work does not possess. For instance, if we include cameras from all directions of the probing vehicle, we will be able to inform the model when there are vehicles moving in the opposite direction. In this case, the model will have knowledge to distinguish the pattern generated from a parked car on the roadside between the pattern received from another moving vehicle on the road.

7.0 Conclusion

In this thesis, we present an innovative passive mobile sensing solution for roadside available parking spot detection. Our approach utilizes smartphone microphones to collect inherently generated signals from moving vehicles, and use these collected signals to distinguish vacant parking space and parked cars. Comparing to existing mobile sensing solutions for roadside parking detection, our approach holds the advantage of mitigating the need for dedicated hardware installation.

To realize our idea, we first conducted an empirical study which provided us in depth understanding of the phenomenon. Base on the knowledge we obtained from our empirical study, we designed an end-to-end system to mitigate the challenges and fulfil our goal to make vacant parking spot prediction. Our system comprises a pre-processing module for audio data stream segmentation, a DNN model for vacant parking spot prediction, and a post-processing module to refine the results for parking information dissemination. To train our model, we created our own dataset using data collected over 2 weeks. Under our final evaluation, our approach demonstrated promising results of our data driven approach with an prediction accuracy of up to 93.51%.

Despite the promising results in this thesis, certain challenges still remain unsolved, leaving space for future work. To approach real-world wide deployment, techniques to handle more possible interferences are needed to improve the system robustness. With the rapid development of computer vision technology, we envision video-guided model to be the next step.

Bibliography

[1] Alessio Bechini, Francesco Marcelloni, and Armando Segatori. Low-effort support to efficient urban parking in a smart city perspective. *Advances onto the Internet of Things: How Ontologies Make the Internet of Things Meaningful*, pages 233–252, 2014.

[2] Birgitta Berglund, Peter Hassmen, and RF Soames Job. Sources and effects of low-frequency noise. *The Journal of the Acoustical Society of America*, 99(5):2985–3002, 1996.

[3] Zhang Bin, Jiang Dalin, Wang Fang, and Wan Tingting. A design of parking space detector based on video image. In *2009 9th International Conference on Electronic Measurement & Instruments*, pages 2–253. IEEE, 2009.

[4] DBL Bong, KC Ting, and KC Lai. Integrated approach in the design of car park occupancy information system (coins). *IAENG International Journal of Computer Science*, 35(1), 2008.

[5] Xiao Chen, Elizeu Santos-Neto, and Matei Ripeanu. Crowdsourcing for on-street smart parking. In *Proceedings of the second ACM international symposium on Design and analysis of intelligent vehicular networks and applications*, pages 1–8, 2012.

[6] Jatuporn Chinrungrueng, Udomporn Sunantachaikul, and Satien Triamlumlerd. Smart parking: An application of optical wireless sensor network. In *2007 International Symposium on Applications and the Internet Workshops*, pages 66–66. IEEE, 2007.

[7] Vladimir Coric and Marco Gruteser. Crowdsensing maps of on-street parking spaces. In *2013 IEEE International Conference on Distributed Computing in Sensor Systems*, pages 115–122. IEEE, 2013.

[8] Andrea David, Klaus Overkamp, and Walter Scheuerer. Event-oriented forecast of the occupancy rate of parking spaces as part of a parking information service. In *Proceedings of the 7th World Congress on Intelligent Systems*, 2000.

[9] Paulo R de Almeida, Jeovane Honório Alves, Luiz S Oliveira, Andre Gustavo Hochuli, João V Fröhlich, and Rodrigo A Krauel. Vehicle occurrence-based parking space detection. *arXiv preprint arXiv:2306.09940*, 2023.

[10] Sergio Di Martino and Luigi Libero Lucio Starace. Vehicular crowd-sensing on complex urban road networks: a case study in the city of porto. *Transportation Research Procedia*, 62:350–357, 2022.

[11] Drivers spend an average of 17 hours a year searching for parking spots. Website.

[12] An Exploratory Study of Parking in the Rosemont Housing Development. Website.

[13] How parking ruined everything. Website.

[14] Car Parking and Traffic Congestion. Website.

[15] Károly Farkas and Imre Lendák. Simulation environment for investigating crowd-sensing based urban parking. In *2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, pages 320–327. IEEE, 2015.

[16] 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*, pages 1–14, 2017.

[17] Sanket Gupte and Mohamed Younis. Participatory-sensing-enabled efficient parking management in modern cities. In *2015 IEEE 40th Conference on Local Computer Networks (LCN)*, pages 241–244. IEEE, 2015.

[18] Baik Hoh, Tingxin Yan, Deepak Ganesan, Kenneth Tracton, Toch Iwuchukwu, and Juong-Sik Lee. Trucentive: A game-theoretic incentive platform for trustworthy mobile crowdsourcing parking services. In *2012 15th International IEEE Conference on Intelligent Transportation Systems*, pages 160–166. IEEE, 2012.

[19] Ho Gi Jung, Young Ha Cho, Pal Joo Yoon, and Jaihie Kim. Scanning laser radar-based target position designation for parking aid system. *IEEE transactions on intelligent transportation systems*, 9(3):406–424, 2008.

[20] Amin Kianpisheh, Norlia Mustaffa, Pakapan Limtrairut, and Pantea Keikhosrokiani. Smart parking system (sps) architecture using ultrasonic detector. *International Journal of Software Engineering and Its Applications*, 6(3):55–58, 2012.

[21] Jacek Kopeckỳ and John Domingue. Parkjam jam: Crowdsourcing parking availability information with linked data. In *The Semantic Web: ESWC 2012 Satellite Events: ESWC 2012 Satellite Events, Heraklion, Crete, Greece, May 27-31, 2012. Revised Selected Papers 9*, pages 381–386. Springer, 2015.

[22] Jean-Gabriel Krieg, Gentian Jakllari, Hadrien Toma, and André-Luc Beylot. Unlocking the smartphone's sensors for smart city parking. *Pervasive and mobile Computing*, 43:78–95, 2018.

[23] Kamlesh Kumar, Vijander Singh, Linesh Raja, and Swami Nisha Bhagirath. A review of parking slot types and their detection techniques for smart cities. *Smart Cities*, 6(5):2639–2660, 2023.

[24] Nikos Larisis, Leonidas Perlepes, Panayiotis Kikiras, and George Stamoulis. U-park: Parking management system based on wireless sensor network technology. In *International conference on sensor technologies and applications (SENSORCOMM)*, pages 170–177, 2012.

[25] Sangwon Lee, Dukhee Yoon, and Amitabha Ghosh. Intelligent parking lot application using wireless sensor networks. In *2008 International symposium on collaborative technologies and systems*, pages 48–57. IEEE, 2008.

[26] Trista Lin, Hervé Rivano, and Frédéric Le Mouël. A survey of smart parking solutions. *IEEE Transactions on Intelligent Transportation Systems*, 18(12):3229–3253, 2017.

[27] A Lorea, V Castelluccio, A Costelli, and M Masoero. A wind-tunnel method for evaluating the aerodynamic noise of cars. *SAE Transactions*, pages 1096–1105, 1986.

[28] 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*, pages 76–85, 2014.

[29] U Männi. Smart sensing and time of arrival based location detection in parking management services. In *2010 12th Biennial Baltic Electronics Conference*, pages 213–214. IEEE, 2010.

[30] 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*, pages 123–136, 2010.

[31] Suhas Mathur, Sanjit Kaul, Marco Gruteser, and Wade Trappe. Parknet: a mobile sensor network for harvesting real time vehicular parking information. In *Proceedings of the 2009 MobiHoc S3 workshop on MobiHoc S3*, pages 25–28, 2009.

[32] Luz-Elena Y Mimbela, Lawrence A Klein, et al. Summary of vehicle detection and surveillance technologies used in intelligent transportation systems. 2007.

[33] Enrique Moguel, Miguel Ángel Preciado, and Juan Carlos Preciado. A smart parking campus: An example of integrating different parking sensing solutions into a single scalable system. *Smart Cities*, 98:29–30, 2014.

[34] Keewook Na, Yungeun Kim, and Hojung Cha. Acoustic sensor network-based parking lot surveillance system. In *Wireless Sensor Networks: 6th European Conference, EWSN 2009, Cork, Ireland, February 11-13, 2009. Proceedings 6*, pages 247–262. Springer, 2009.

[35] 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*, pages 963–973, 2014.

[36] 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*, pages 75–86, 2013.

[37] Vijay Paidi, Hasan Fleyeh, Johan Håkansson, and Roger G Nyberg. Smart parking sensors, technologies and applications for open parking lots: a review. *IET Intelligent Transport Systems*, 12(8):735–741, 2018.

[38] Mikko Rinne, Seppo Törmä, and D Kratinov. Mobile crowdsensing of parking space using geofencing and activity recognition. In *10th ITS European Congress, Helsinki, Finland*, pages 16–19, 2014.

[39] Rosario Salpietro, Luca Bedogni, Marco Di Felice, and Luciano Bononi. Park here! a smart parking system based on smartphones' embedded sensors and short range communication technologies. In *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, pages 18–23. IEEE, 2015.

[40] Xavier Sevillano, Elena Màrmol, and Virginia Fernandez-Arguedas. Towards smart traffic management systems: Vacant on-street parking spot detection based on video

analytics. In *17th International Conference on Information Fusion (FUSION)*, pages 1–8. IEEE, 2014.

[41] SF-Park. Website.

[42] sfpark-wik. Website.

[43] Donald Shoup. Cruising for parking. *Access Magazine*, 1(30):16–23, 2007.

[44] Leon Stenneth, Ouri Wolfson, Bo Xu, and S Yu Philip. Phonepark: Street parking using mobile phones. In *2012 IEEE 13th international conference on mobile data management*, pages 278–279. IEEE, 2012.

[45] Nguyen Thai-Nghe and Nguyen Chi-Ngon. An approach for building an intelligent parking support system. In *Proceedings of the 5th Symposium on Information and Communication Technology*, pages 192–201, 2014.

[46] Vasilis Verroios, Vasilis Efstathiou, and Alex Delis. Reaching available public parking spaces in urban environments using ad hoc networking. In *2011 IEEE 12th International Conference on Mobile Data Management*, volume 1, pages 141–151. IEEE, 2011.

[47] Jesus Villalobos, Bereket Kifle, Derek Riley, and Jesús Ubaldo Quevedo-Torrero. Crowdsourcing automobile parking availability sensing using mobile phones. In *UWM Undergraduate Research Symposium*, pages 1–7, 2015.

[48] Félix Jesús Villanueva, David Villa, Maria José Santofimia, Jesus Barba, and Juan Carlos Lopez. Crowdsensing smart city parking monitoring. In *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, pages 751–756. IEEE, 2015.

[49] Gongjun Yan, Weiming Yang, Danda B Rawat, and Stephan Olariu. Smartparking: A secure and intelligent parking system. *IEEE intelligent transportation systems magazine*, 3(1):18–30, 2011.

[50] Bin Yang, Nicolas Fantini, and Christian S Jensen. ipark: Identifying parking spaces from trajectories. In *Proceedings of the 16th International Conference on Extending Database Technology*, pages 705–708, 2013.

[51] Mingmin Zhao, Ruipeng Gao, Jiaxu Zhu, Tao Ye, Fan Ye, Yizhou Wang, Kaigui Bian, Guojie Luo, and Ming Zhang. Veloc: Finding your car in the parking lot. In *Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems*, pages 346–347, 2014.

[52] Wan Chen Zhao. Parking management system with rfid and sensor networks. *Applied Mechanics and Materials*, 198:1690–1696, 2012.