MagnoPark – Locating On-Street Parking Spaces using Magnetometer-based Pedestrians' Smartphones

Maryam Arab
Department Of Computer Science
Old Dominion University, Norfolk, USA
Email: marab@cs.odu.edu

Tamer Nadeem
Department Of Computer Science
Old Dominion University, Norfolk, USA
Email: nadeem@cs.odu.edu

***Abstract*—In heavily congested urban areas, the rapid growth of population is becoming more and more of an issue. Affected cities quickly demand solutions to areas such as quality of life, waste management, public transportation, and accessibility to main resources. However, since the number of impacted areas of population growth is endless, we focus on public parking. Studies show that drivers spend a large portion of their travel time locating vacant parking spots. For this reason, we present MagnoPark, a crowdsourced approach to identifying unoccupied spots accessible to the general public, who are typically free. MagnoPark is a smartphone based sensing solution that detects empty parking spots using internal sensors of cellphones. While a pedestrian is walking on the sidewalk, we exploit magnetometer changes near metal objects in identifying where cars are located. The amplitude and rate of change shift dramatically when approaching or passing cars that are parked beside the street, giving us a great platform towards solving the defined problem. With empirical evaluation, we show that not only is our solution a notable step towards economical parking management but also significantly efficient and as accurate as traditional sensor-based parking solutions.**

I. Introduction

In large cities, a parking space is both an expensive and a hard to find resource. On a daily basis, a large portion of vehicles on road in urban environments constitute to those seeking a parking spot [1]. While the impact is sporadic in nature, at times heavily influenced by the geographic location or the contextual side of its environment, it is an obvious challenge. According to [2], finding a space to park can take as long as 15 minutes on average in major metropolitan areas. The cause for extraneous search is primarily developed due to: i) drivers tend to search for spots by preference in which free on-street parking closest to a particular destination is of ideal value, and ii) drivers' tendency leads them, when all ideal spots are occupied, towards waiting and actively seeking alternative locations. This issue of inability to know where else to park promotes misuse of driver time, which increases traffic congestion and creates health issues due to the emissions released by vehicles [2], [3].

Since this is certainly not a new issue, and is only increasing level of inconvenience, dense urban areas are beginning to invest heavily towards implementing potential solutions. Some of those include Fastprk [3] and SENSIT [4], which are both sensor based systems for detecting when parking spots are occupied. While they both require hardware equipment to fully function, they integrate with public payment and notification systems to help streamline the parking process. Both of these solutions aim to identify vacant spots, guide drivers towards potential locations, and increase driver satisfaction and the overall city management. Fastprk implementation claims a 35% improvement in the time needed to park [2], while SENSIT claims both a 64% reduction in park violations and a decrease in space occupancy [4].

While this is a good approach for locations which generate revenue, i.e., paid parking on popular streets, cities lack similar technology for free curb side parking. Investing into previously mentioned solutions is still an option for well-funded cities. However, for those that have a limited budget, covering all potential streets of interest can quickly grow into financial exhaustion. Take for example a city like Chicago or New York, where the number of crowded streets is potentially endless, demanding a large portion of the cities budget for a complete conversion.

Recent survey study shows that about %80 of US population own a smartphone in 2016 [5]. This statistic motivates us to develop MagnoPark; a crowdsourcing approach that utilizes on-street pedestrians and their smartphones to identifying unoccupied on-street parking spots. MagnoPark is a smartphone based sensing solution that leverages the magnetometer sensor on smartphones to detect empty parking spots. While a pedestrian is walking on the sidewalk, we exploit the change in magnetometer sensors when approaching or passing by vehicles that are parked on street side to detect where vehicles are located and consequently, identifying the on-street available parking spots. We evaluate MagnoPark on several streets in downtown under different conditions and users with different walking speeds. Results show that MagnoPark detects available parking spots with more than %95 accuracy.

We summarize the main contribution of this paper as follows:

- Develop a high-accuracy classification scheme that utilizes smartphone magnetometer sensor in detecting on-street parked vehicles and other on-street metal objects such as light poles. To our knowledge, this is the first project that utilizes smartphone magnetometer sensor in detecting vehicles.
- Develop and implement MagnoPark; a low-cost high-accuracy crowdsourcing approach for detecting on-street parking spots using smartphones carried by on-street pedestrians.
- Evaluate MagnoPark on several streets under different conditions and users with different walking habits and speeds.

The rest of this paper is organized as follows. Section II describes briefly the background and the related work. We present MagnoPark architecture in Section III. In section IV, we describe the details of MagnoPark modules and components. Evaluation of MagnoPark is described in Section V. Section VI concludes the paper and highlights our future work.

II. Background and related works

A. Background

Magnetometers are very sensitive to soft and hard iron. This sensitivity is caused by the distortion in earths magnetic field. Magnetometers sense the change in earth's magnetic field that is caused by a metallic object. The reason for this distortion is that the magnetic field flows more easily in the ferromagnetic materials than air. This effect causes earth's magnetic field lines to be bent quite a bit in the presence of any metal object including vehicles [6]. Figure 1 (top) shows earth's magnetic field that are almost parallel by

* This research is partially supported by the National Science Foundation under grant CNS-1454285.

978-1-5090-6599-8/17/$31.00 ©2017 IEEE

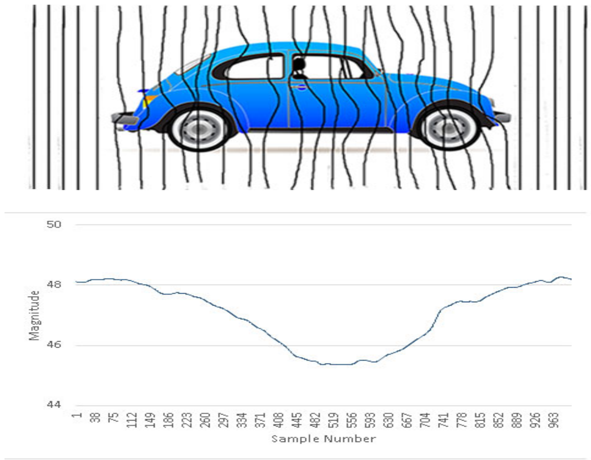


Fig. 1: Earth magnetic field distortion in the presence of a vehicle (top figure) and variations in Magnetometer readings corresponding to magnetic field distortion (bottom figure).

vertical lines in which the presence of a vehicle causes these parallel lines to be bent and distorted.

Magnetic fields sensing has expanded vastly as many magnetic sensors are used to detect the strength, direction and distortion of not only earth's magnetic field, but also fields generated by electric currents, permanent magnets, and vehicle magnetic field disturbance. Magnetic sensors are able to detect these changes without any physical contact.

Many navigation control systems have an eye on this feature to correct the magnetic deviation caused by hard iron and soft iron to reach an accurate tracking for both under water and out of water vehicles. Strong algorithms including Kalman filter [7] are used in these systems to correct the distortion that is caused by any kind of hard or soft iron objects in the earth magnetic field. In addition for tracking purpose, portable magnetic sensor systems are designed and developed to be used beside the roads for vehicle counting and classification and also for speed measurements [8].

Giving these characteristics and by leveraging the smartphone magnetometer sensor, we observed from our experiments that smartphone magnetometer sensor experiences significant variations in readings in the presence or absence of a vehicle as shown in Figure 1(bottom). Consequently, we utilize this observation in developing MagnoPark to identify and locate available on-street parking spots to later notify nearby drivers who are looking for parking spaces.

B. Related Work

In this section, we overview parking detection systems and summarize their features. By comparing MagnoPark with these parking detection systems, we will show our contributions of our work in this article.

RFID technology is one of the popular ways researchers are battling smart parking, where small instruments are installed in each vehicle to communicate with a base station. By using this approach, individuals can be identified by their device, and management applications can get a head count as to how many spots are vacant or filled [9]. While this system decreases waiting times and traffic jams, it comes with three main disadvantages. First, a system that requires implementation in all vehicles is a rather costly solution for both the driver and those maintaining the evolution of the proposed technology. Second, this solution can be very error prone in dense areas, as multi-broadcast collisions can prevent several vehicles of entering a parking lot simultaneously. Lastly, security issues can arise as a limited amount of preventative measures are being taken towards protecting devices identifiers from spoofing.

Unlike RFID solutions, there are those leveraging light as a medium for parking identification. By measuring the distance that vehicles cover as they travel throughout a particular area, similar solutions can be implemented. One of those can be seen in [10], where the authors develop a LIDAR system consisting of light sensors who tracking movement of all entered vehicles. At the end of a travel cycle, a map is generated for a particular path, and estimation can be made as to what spots are no longer vacant. A similar approach is taken by [11], which while very accurate is a rather expensive solution requiring a plethora of equipment to configure.Other solutions, alike those outlined in [12], entail video and image processing, scattered transmitter nodes for information relay, ultrasonic waves and microwaves towards vehicle localization.

SFpark [13] is a parking management system, which adopts a wireless sensor network structure. The SFpark pilot deployment installed 11700 magnetometer sensors and 300 pole-mounted mesh nodes for 8000 parking spaces in California (for each parking space,one or two sensors are installed.) The data from parking sensors is fed to a wireless mesh network and will be pushed to a data warehouse. Although this solution is very simple in finding on-street parking spots, which will helps in reducing the traffic congestion, the cost of this parking system is very high. Fastpark [2] is another similar Magnetic solution allowing a Town Council to turn its town into a smart city. Both the systems are very costly and require digging roads to embed at least one magnetic sensor per spot.

ParkNet [14] is another similar work, a mobile on-street system comprising vehicles which collect parking space information while driving by. Each ParkNet vehicle should be equipped with a GPS receiver and an ultrasonic rangefinder to determine parking spot occupancy. The data is aggregated at a central server, which builds a real-time map of parking availability and could provide this information to clients that query the system in search of parking spots. In order to achieve improved location accuracy, authors utilize an environmental fingerprinting approach and use objects on the street to correct GPS errors.

Similar to MagnoPark, current research attempts to identify new means of tracking vehicles and preventing side effects of congested parking in crowded cities. However, a line between accuracy and cost is quickly expanding, in which the financial aspect dictates the level of performance. Our research, on the other hand, is distinctively different as MagnoPark is both accurate and considerably less costly than current solutions.

III. MagnoPark Architecture

Figure 2 shows the architecture of MagnoPark, which consists of three main components: Pedestrian component, Cloud server component, and Driver component. MagnoPark leverages the mobility of pedestrians on-street sidewalks and opportunistically: i) collect several sensor data from their smartphones, ii) process these the data to detect parked vehicles, and then iii) assess the on-street parking spots conditions and make it available to drivers nearby.

We have developed MagnoPark-Ped smartphone application that runs on pedestrian smartphones, and executes the different modules of MagnoPark pedestrian component. Once the application starts, it collects accelerometer, gyroscope, magnetometer, and GPS data. Magnetometer sensor readings are the main data that is used by MagnoPark to extract several features, as we will describe later, to

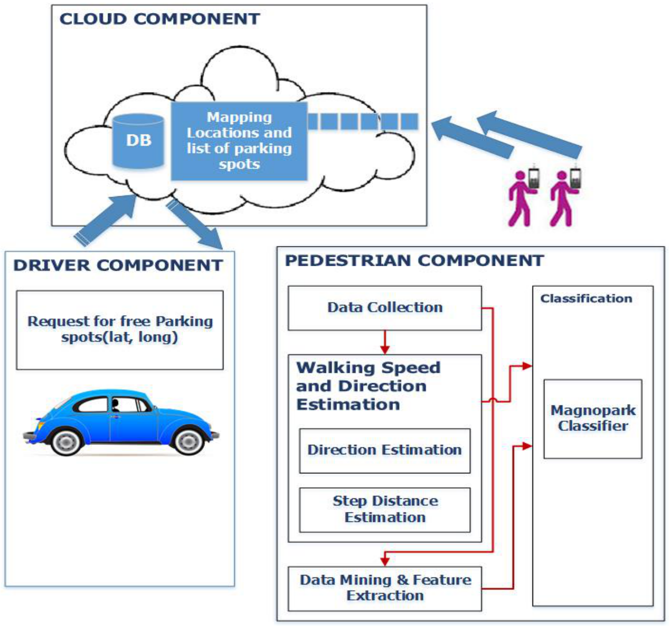


Fig. 2: Architecture overview of the different components of Magnopark system.

detect the on-street parked vehicles. Note that, to make MagnoPark independent of smartphone orientation, we use the magnitude of the magnetometer 3-axis vector readings in our calculations. GPS data is used by MagnoPark cloud server to map the data from each pedestrian to the corresponding street and, consequently, to map the detected parked vehicles to the different parking spots on the street. Accelerometer data in company with gyroscope data are used to detect the walking status of pedestrians (e.g., walking, standing). In addition, gyroscope and accelerometer sensors are used to track individual pedestrian/user steps to be able to calculate her walking speed. Since different users have different walking speeds, it is important to consider individual user speed in order to be able to correctly correlate the collected samples from this user to on-street parking spots. For example, a faster pedestrian will collect fewer numbers of samples than a slower pedestrian of the same-parked vehicle. In addition, we use gyroscope to detect the change in user direction, which again helps to correctly correlate the collected data to the corresponding parking spots. For example, if the a user reverse his direction, this will help us to avoid duplicate detections of the same parked vehicles. With all these sensors data, we design and develop a classification algorithm to classify and detect parked vehicles on the street from collected data. After identifying the locations of parked vehicles, we would be able to identify the available parking spots along the corresponding street.

Classified data by MagnoPark-Ped application is uploaded to MagnoPark cloud component. This particular component is responsible for storing, processing, and mapping all pedestrians' classified data. It is responsible to correlate available on-street parking spots to physical street locations in order to facilitate this information for nearby drivers seeking for parking spots. It is important to note that only parking spots information (i.e., occupied or available) along with corresponding GPS information are sent to and stored by the cloud component. No additional user personalized information (e.g., user ID) or phone information (e.g., wireless MAC address, SIM card information) are stored in order to preserve the privacy of participating users. In other words, data uploaded can't be correlated to any other stored data in the cloud and consequently no data stored can be used to identify or to be correlated to any participating user.

The Driver component is a simple component that consists of one module in which the driver uses to send a parking request along with his location the to the cloud server for the nearest parking spots. In response, the cloud server sends back a local map of nearby area with current available parking spaces.

IV. MagnoPark Components

In the following subsections, we describe each the the three MagnoPark components in more details.

A. *Pedestrian Component*

In MagnoPark, pedestrian component consists of the following four main modules that are shown in figure 3:

1) ***Data Collection****:* We developed MagnoPark-Ped android application to collect and analyze data exerted by smartphones. The system collects readings from accelerometer, gyroscope, magnetometer, and GPS. In order to detect a vehicle, we resort to the variations in the magnitude of the 3-axes magnetometer sensor as pedestrians/users are walking on sidewalks and passing by on-street parked vehicles. Therefore, our calculation is independent of user smartphone orientation. This point makes MagnoPark more practical since there is no specific requirement from user to hold or carry their smartphones in special way.

In this project, the application collects magnetometer data with frequency rate of 50Hz. This rate is high enough to detect the slightest changes in the magnitude value. We experimented collecting magnetometer data with different frequency rates (i.e., 20Hz and 100Hz) and did not find significant difference. We also use gyroscope and accelerometer data to calculate the speed and direction of walking. We use the walking speed to calculate the corresponding optimum searching window size, as we will describe later, that will be used in detecting parked vehicles and calculating their length. Walking speed is also used to estimate the distance between consecutive parked vehicles and, consequently, to estimate the number of available parking spots between consecutive parked vehicles. Moreover, GPS data is collected to upload to cloud server the latitude and longitude of detected parking spots in order to map these spots to the corresponding street spaces.

2) ***Walking Speed and Direction Estimation****:* This module consists of two main sub-modules:

Distance Estimation: This module estimates the distance traversed by pedestrian at each step. We use accelerometer and gyroscope sensors of the user smartphone to detect and track user steps. We utilize the changes in gyroscope sensor to activate this module once the user starts moving. Whenever the gyroscope sensor readings reach above a certain threshold (in our case 0.3), we infer the user movement. Detecting the user movement triggers the application to start capturing accelerometer, magnetometer and GPS data. Since the user step length is, to some extent, proportional to his speed [15], we consider two parameters m_{max} and m_{min} to represent the maximum and minimum length of a user step in terms of number of samples:

$$m_{max} = s_{max}/v_{max} * f_a$$

$$m_{min} = s_{min}/v_{min} * f_a$$

where s_{max}, and s_{min} are the maximum and minimum length of a user step, v_{max} and v_{min} are the maximum and minimum walking speed of a user, and f_a is the frequency of the accelerometer

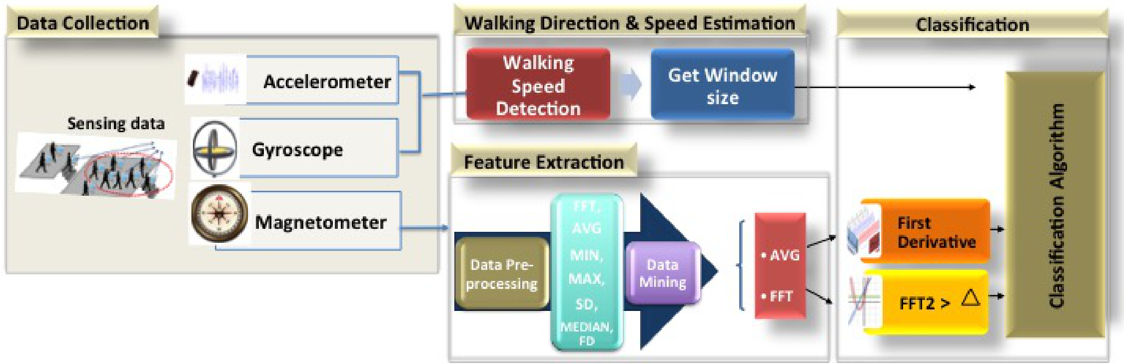


Fig. 3: The modules of Pedestrian Component of MagnoPark

samples in the smart phone [16]. After collecting m_{max} rows of accelerometer samples, we calculate the magnitude of the accelerometer 3-axis vector of each sample, to be sure that it is independent of the orientation of the smartphone. Next, we apply Finite Impulse Response low pass filter to remove any noise and then followed by a normalization step. Then, we feed the samples to Dynamic Time Wrapping (DTW) algorithm [17]. DTW uses a predefined pattern of a step to detect whether there is a step defined between the next m_{min} and m_{max} samples. If a step is detected, then we shift the next processing samples window by the number of samples of the detect step. If not, we shift by one sample. The advantage of DTW is that it can detect walkings steps for different walking speeds. We use the commonly used personalized step length model [17] to estimate the step size of the user and consequently the user walking speed[1]. For more details, please refer to [16]. It is very important to estimate user walking speed to correctly correlate the magnetometer collected samples from each user to on-street parking spots. For example, it is essential to differentiate between when a user stands still beside an available parking spot and when he is walking beside a row of available spots.

Direction Estimation: We need to track the walking direction of each user in order to avoid the erroneous in detecting parking spots. For example, if a user reverses his direction, this will help us to avoid duplicate detections of the same parked vehicles. In order to estimate the user direction, we have to align three different coordinate systems: smartphone coordinate system, user walking coordinate system, and global coordinate system [16]. As the global coordination is fixed, we align the other two systems to it. After the alignment, the highest variation of the linear acceleration readings will be aligned with the user walking direction. We apply Principal Component Analysis (PCA) analysis to find out the walking direction of the user. For more detailed description about walking and direction estimation, please refer to [16].

*3) **Data Mining and Feature Extraction**:* In this module, our feature extraction relies on both motion sensors and magnetometer sensor readings. The presence of on-street parked vehicles will be reflected in changes in the magnetometer readings. These changes as mentioned before are due to the earth's magnetic field distortion near metallic objects.

Our main goal is that the selected features should be able to differentiate between parked vehicles and no vehicles as well as to be able to differentiate between vehicles and any other on-street metal objects such as trash bins and light poles. In this subsection, we first describe the important issues that we consider in our model; followed by the steps we used to extract effective features from magnetometer sensor readings.

User motion is an important factor in detecting parked vehicles using the magnetometer in our system. Our experiments, as described earlier (Figure 1 (bottom)), show that the magnitude of the raw 3-axis magnetometer data changes significantly as the user is approaching and passing by a parked vehicle. On the other hand, this magnitude remains almost stable when there is no vehicle nearby the user. Although the magnetometer magnitude has higher values when the user is standing next to a vehicle than there in no vehicle, our systems relies more on the relative changes in the readings rather than the absolute values. Therefore, collection of data is only executed as the user is in motion, which is detected as described earlier.

Walking speed is another important factor in differentiating between vehicles of different sizes, metal bars, or any other on-street metal objects. By estimating the walking speed as described before, we can estimate the vehicle length by correlating the amount of change in the magnitude curve to the walking speed. Detecting vehicle sizes and differentiating them from other metal objects is very essential for high accuracy performance of MagnoPark.

In processing the collected sensor data, we first divide the collected samples into sequence of non-overlapped one-second periods we refer to as data "*epochs*". As we assumed to collect sensor readings with frequency of 50Hz, each epoch will consists of 50 samples in which each sample includes magnitude ($\sqrt{x^2 + y^2 + z^2}$) of the raw 3-axis magnetometer, 3-axis accelerometer, 3-axis gyroscope, GPS data, and timestamp[2]. As we described earlier that our MagnoPark is mainly based on the magnetometer sensor in detecting parked vehicles, we limit our extracted features to the magnetometer readings. In doing this, for each epoch we extract a set of features based only on the magnetometer readings. One of the features we extract is the magnitude average (Avg) in which we calculate the average of magnetometer magnitude values of all the samples per epoch as shown in step #1 in Figure 4. We use Avg_i to refer to a magnitude average that is corresponding to i^{th} epoch. Although this value (and consequently the magnetometer raw readings) highly depends on the distance between the user and the vehicle, from experiments we found that the variations in these values for different positions of the user on the sidewalk is not that significant as long as the distance between the user and the vehicle is less than three meters, which fits with typical width of most of the street sidewalks.

In addition to the magnitude average, we calculate the minimum,

[1]We build the personalized model for each user of MagnoPark-Ped application at the time the user runs registers with MagnoPark system for the first time.

[2]For sensors with lower sampling frequency such as GPS, we use same values of the sensor for some of the consecutive samples

maximum, median, and standard deviation of the magnitude samples of each epoch. In addition, to capture the changes in magnetometer reading as the user approach/passing by a vehicle, we calculate the first derivatives over the magnitude averages (Avg) of consecutive epochs. We refer to these derivatives as FD. To calculate this feature, we first apply low pass filter on the calculated magnitude averages in order to reduce noise and redundancy and, then, we normalize these average values. We refer to the normalized value of a magnitude average Avg_i as N_i as shown in step #2 in Figure 4. Since vehicle presence is the source for variations in magnitude values, we calculate the first derivatives of the normalized data using equation 1:

$$FD_i = \frac{N_i - N_{i-1}}{t_i - t_{i-1}} \tag{1}$$

where FD_i is the calculate first derivative of magnitude average corresponding to i^{th} epoch. Since each normalized magnitude value is corresponding to one epoch and since each epoch is corresponding to one sec, $dt = t_i - t_{i-1}$ is equal to 1. Step #3 in Figure 4 shows the calculation of FD.

In addition to the above six basic features, we use Fast Fourier Transform (FFT) to capture and extract additional features. In doing so, we apply Fast Fourier Transform magnetometer magnitudes of each epoch to generate the frequency spectrum of the samples, and then we select the first ten frequencies ($FFT1$ to $FFT10$) as another ten features of magnetometer for each epoch. Giving that, for each epoch of magnetometer magnitudes, we extract the sixteen features that will be evaluated in selecting the most effective features for our system MagnoPark.

*4) **Data Classification:*** In this subsection we describe the training and testing of the classifier MagnoPark uses in detecting on-street parked vehicles. As we describe in the next section, we used a group of users to walk around large number of streets to collect both i) the readings of their smartphones sensors, and ii) the ground truth data about parked vehicles. We split the collected data in to two sets; training data and testing data. We used 80% of the collected data and the corresponding extracted features as the training set to build the classification model, while the remaining 20% is used as a testing data to evaluate our developed classifier[3]. In training the classification model, we use several data mining tools such as Weka and Microsoft Excel SQL Data Mining tools combined with Decision Tree and Naive Bayes algorithms to extract the most effective features that have the highest effect on our detection. The outcome of data mining using the training data set showed that the there are two prominent features among the sixteen extracted features in which are critical in our classification model as follows:

I) $FFT2$ - Among the first ten extracted FFT parameters by applying Fast Fourier Transform (FFT) on the magnetometer magnitudes in each epoch, the second value of these FFT values, which we refer to by $FFT2$, is above a certain threshold value ($fft - threshold$) when the user is passing by vehicles. From our experiments, we found that the value of this threshold is 2.24

II) First Derivatives of Magnitude Averages (FD) - The second and very effective feature in our classification model is the first derivate that is calculated over the magnitude averages of consecutive epochs.

Using these two main features, we build our classifier model to classify whether these features of each epoch is corresponding to a

[3]We used another independent set of users to collect a new data set to evaluate our MagnoPark as we describe in Performance Evaluation section

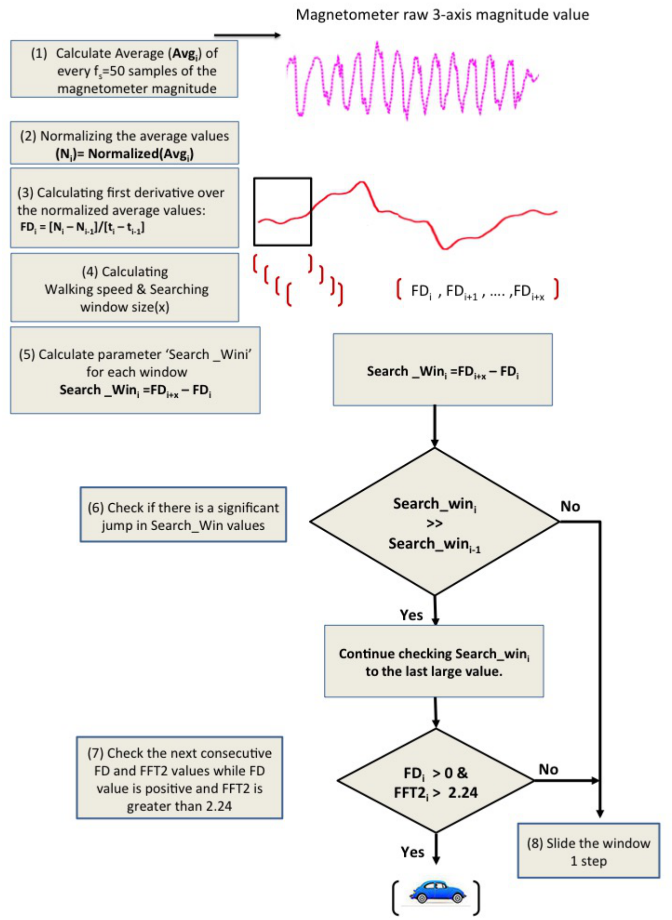


Fig. 4: MagnoPark classification process for detecting on-street parked vehicles.

presence of a parked vehicle. Steps #4-#8 of Figure 4 shows the steps of the classification and are summarized in the following main steps:

1) Steps #4 - Specify the searching window size (x): Based on the calculated pedestrian walking speed as described earlier, we specify a window for our classifier. This window size is related to the length of a data set that is correlated to the presence of a regular vehicle when the user is passing by. For example, if the user walks with a rate of 0.5 meter per seconds beside a 3-meter length vehicle, the user needs 6 seconds to pass the vehicle completely. Therefore, in this case, the window size is set to be 6. With the normal walking rate, which is equal to 1 meter per seconds, the same length will take 3 seconds, which corresponds to a window of size 3. We refer to the searching window size as x, which should be approximated to multiple of epochs. Note that this window size changes with the change in the user walking speed.
2) Step #5 - Calculate Parameter $Search\_Win_i$: Based on the calculated value of x in the previous step, we calculate the variations of FD values for overlapping windows of the data set as follows:
$for\; i = 0\; to\; n$
$Search\_Win_i = FD_{i+x} - FD_i$
in which n is the total number of epochs. From experiments, this parameter has small values (e.g., less than 1). Once the user approach the vehicle boundary, this parameter starts to have consecutive large positive values comparing to the previous

data.

3) Step #6: In this step as shown in figure 4, we check if there is a significant jump in the value of $Search\_Win_i$ compared to $Search\_Win_{i-1}$. As described, large positive values of this parameter indicates approaching a vehicle and that the last large positive value marks the beginning of the vehicle. Therefore, we use this parameter as a "detection boundary" of the vehicle in which the classification algorithm starts to check the consecutive FD and $FFT2$ values (in the next step) after hitting the last large value in the detection boundary.
4) Step #7: After the model detects the last large positive in the detection boundary, we check the values of consecutive FDs and $FFT2$ values. If the value of FD is positive and the value of $FFT2$ is greater than 2.24, we label this epoch with vehicle presence (C). Otherwise, we mark it as no vehicle (nc).
5) Step #8: Advance and repeat by going to step 6.

As shown, our classifications model combine both the two features $FFT2$ and FD in a binary classification to classify whether a vehicle is presence (C) or no vehicle (nc) for each epoch. Then, a parked vehicle is detected if three or more consecutive epochs are labeled with vehicle presence (C). Otherwise, no vehicle is detected.

B. Cloud Server Component

In MagnoPark, once the classifier classifies an on-street parked vehicle, the pedestrian component uploads this information with the corresponding GPS and timestamp to the cloud server. Note that that to minimize the number of uploads to save power, the pedestrian component could upload the detected parked vehicles once the user reached to the end of the current street segment. The typical scenario of a pedestrian is that he gets off the street sidewalk once he reaches its end, cross to the next segment, and then gets on the sidewalk of the next street segment. We use the accelerometer sensor to detect transitions of stepping off and on of the street sidewalk.

One of the main responsibilities of cloud server component is to match the traversed path to the street map of the area that could be extracted from Google Maps. In order to determine which street the pedestrian is walking on, we use Map Matching approach that is used to detect the street using smartphone sensors data. For this purpose, the street will consist of series of points in which each point consists of a time-stamp latitude-longitude pair of the detected parked vehicle. Work in [18] presents a new map matching algorithm based on the Hidden Markov Model (HMM) which map the user path to the Map. Combining the detected parking spot and Map Matching approach, the process of mapping each street location coordinates with a list of available parking spots is performed on the cloud server. This data will be updated when any changes is pushed to the server by other pedestrians.

C. Driver Component

As we mentioned earlier, when a driver is looking for an available parking spot, he sends a request (using his MagnoPark application for drivers) to the cloud server accompanied with his GPS location. In response, the cloud server sends back to the driver a list of available parking spots that could be easily mapped on the driver device local map. Then, the driver chooses from the available spots and gets the directions to that parking space.

V. PERFROMANCE EVALUATION

In this section, we describe how we conducted our experiments in order to i) collect ground truth data to build our classifier component described in Section IV, and ii) evaluate MagnoPark performance under different scenarios and conditions.

A. Experiment Configurations

To collect our ground truth data and to conduct our experiments, we developed an Android application that collects the magnetometer readings at frequency of 50 samples per second. In order to collect the ground truth data to track and log the times in which the user is passing by a vehicle without disturbing user walk, we developed single tab and double tabs features in our application in which the user tabs the phone once he begins to pass by a vehicle and double tab the phone once he finishes passing the vehicle. Another important issue that we considered in our experiments is to differentiate between a vehicle and other metal objects on the street such as trash bins or light poles. In order to do so, we added triple tap feature that the user will use once he is passing by any of these non-vehicle objects on the street. These collected time logs are used as the ground truth for correlating between the collected magnetometer readings and vehicle's presence to train and build our classifier.

In conducting all of our experiments, we used ten graduate students as our users of mixed genders and with different walking habits and speeds. We asked our users to hold or carry their smartphones in the neutral as they use to do such as holding it in hand or carrying it in their pockets or bags. In addition, we asked our users to walk normally without any constraints or limitations. Note that we did not depend on smartphone camera for our ground truth data since it will prevent users from using their phones in a normal way as they typically do. We also asked the users to repeat their walks several times (i.e., average of five runs). We also asked different users to repeat the same experiment in the same place and situation with their different walking habits and speeds.

To evaluate the device Independence accuracy of MagnoPark, we conducted our experiments using different phones including Samsung Galaxy 5 running Android Lollipop OS and LG Nexus 4 E960. In addition, we conducted our experiments in different locations including parking lots, private streets, crowded streets, and downtown streets with shopping stores.

B. Experiment Scenarios

We evaluated MagnoPark under different scenarios and conditions. We classify the scenarios into two main classes: a) controlled/small-scale scenarios, and b:) realistic/large-scale scenarios as follows:

- **Controlled/Small-scale Scenarios:**
  The objective of these scenarios to train and tune MagnoPark classifier and evaluate the performance of MagnoPark under specific conditions and configurations. These scenarios include: i) *No_Vehicle* scenario in which we evaluate the magnetometer sensors on streets with both significant traffic of vehicles and with no traffic at all. On either case, no vehicle is allowed to park by the street. The objective is of this scenario to evaluate the impact of street traffic on the performance of MagnoPark. ii) *Multiple_Vehicles_with_Spaces* in which conducted experiments to evaluate whether MagnoPark will be able to differentiate between consecutive parked vehicles from parked bus/truck and detect the number of available parking spots between parked vehicles. and iii) *Mix_Vehicles_with_Metal_Objects* in which we evaluate whether MagnoPark is able to differentiate between vehicles and other on-street metal objects such as trash bins and light poles.
- **Realistic/Large-scale Scenarios:**
  After we trained and tuned MagnoPark, we conducted several experiments on down town streets with shopping stores and significant traffic of vehicles and pedestrians at different times using several users with different walking styles and speeds.

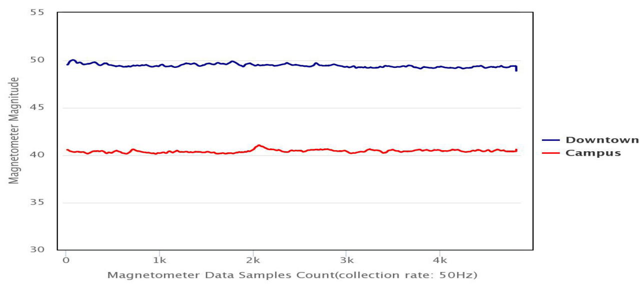


Fig. 5: Changes in Magnetometer readings on campus street and downtown street with no parked vehicles.

In these experiments, we asked different users to repeat the same experiment on the same street and time. For each of these experiments, only one user is asked to collect the ground truth while the others where asked to just walk down the street.

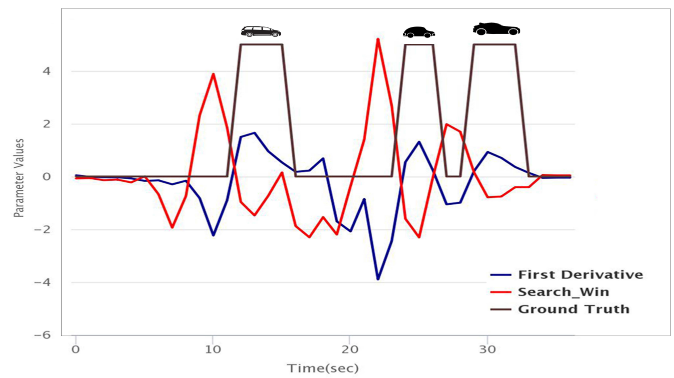
C. Experiment Results

In this subsection, we present the results of different scenarios described above.

1) Controlled/Small-scale Scenarios: We used the described above *No_Vehicle* scenario to evaluate the sensitivity of smartphone magnetometer sensors when there there is no parked vehicle. We conducted set of experiments on our campus streets as well as down town streets with shopping centers in which there is no parked vehicle on any of these streets. Figure 5 plots the magnitude of the raw 3 axes magnetometer data for two of these experiments corresponding to on campus street and a downtown street. As shown, the magnitude in any of these scenarios stays almost stable while the higher value of the magnetometer at downtown streets is due to the on-streets stores with their metal construction. Both plots show that on-street vehicle traffics do not have any significant impact on the smartphone magnetometer sensor and consequently on MagnoPark performance. This is due to that the magnetometer readings are very dependent to the distance between the smartphone and the vehicle and the longer the distance, as in the case of the on-street traffic, the less the sensitivity the magnetometer sensor.

To evaluate the sensitivity of MagnoPark to differentiate between multiple consecutive vehicles versus large vehicles (e.g., trucks, buses), we experimented with different configurations of *Multiple_Vehicles_with_Spaces* scenario. Figure Figure 6(a) shows one of these configurations in which we have one SUV vehicle, a space and then two consecutive vehicles. Figure 6(b) shows the logged ground truth corresponding to this configurations and both FD and $Search\_Win$ parameters (features) described in Section IV.A, which are corresponding to one of the experiments. A shown in Figure 6(b), the two consecutive bumps in FD curve clearly indicates that Magnopark is able to detect consecutive parked vehicles. To better understand how MagnoPark classifier works to detect vehicles, we use Table I that shows the main parameters we used in our classifier as the user passes by the SUV vehicle, the available space, and then the first vehicle of the two consecutive vehicles. In this table, "$Detection$" column represents the detected values (i.e. C (vehicle detected and highlighted in green) or nc (no-vehicle detected)) using our classifier algorithm (Section IV.A). On the other hand, "GT" column represents the ground-truth values (i.e., 1 (vehicle and highlighted in blue) or 0 (no-vehicle)). The algorithm looks for values in "$Search_Win$" column that have higher values comparing to the previous values. These corresponding cells are highlighted in gray color. These values could be used as indicators that the user is

(a)



(b)

Fig. 6: (a) Vehicle configuration for one of the experiments that consists of SUV vehicle, a space, and then two consecutive vehicles. (b) The logged ground truth corresponding to vehicles configuration in (a) and both FD and $Search\_Win$ parameters used in our classifier (described in Section IV.A).

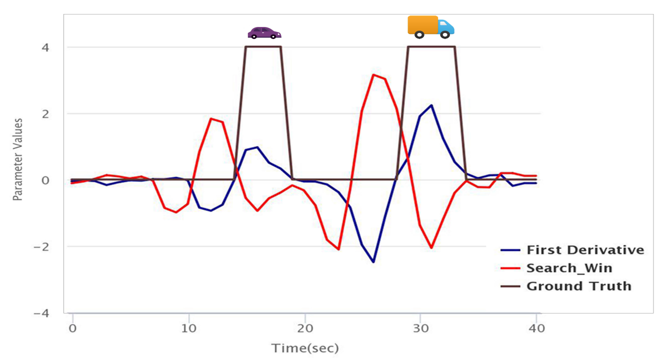


Fig. 7: The logged ground truth and both FD and $Search\_Win$ parameters as the user passes by a vehicle and a small truck

approaching a vehicle. To confirm that the changes in these cells' values is due to approaching a vehicle, the algorithm examines the corresponding values of FD parameter (i.e., the first derivative of magnitude), which are highlighted in green under "FD" column of the table. While those values are positive and greater than a specific threshold (we choose 0.1 as the threshold in this work), they are used to indicating the presence of a vehicle as shown in "$Detection$" column. Note that, for most of the readings of this table, the searching window size (x) has the value of 3.

We run another set of experiments to evaluate the sensitivity of MagnoPark to consecutive vehicles versus large vehicles such as buses, trucks, or vans. Figure 7 shows the ground truth as well as

TABLE I: Calculated features described in Section IV.A for the experiment when the user passes by two of the vehicles in Figure 6(a)

Avg	FD	Search_Win	FFT2	GT	Detection
48.497	0.066	-0.436	2.138	0	nc
48.563	0.115	-0.908	2.154	0	nc
48.679	-0.169	-1.005	2.166	0	nc
48.510	-0.370	1.060	2.143	0	nc
48.140	-0.793	4.884	2.110	0	nc
47.347	-1.174	4.279	2.021	0	nc
46.173	0.691	-1.814	2.290	0	C
46.864	4.090	-6.191	2.463	1	C
50.954	3.105	-4.195	3.079	1	C
54.060	-1.123	0.493	3.310	1	nc
52.936	-2.101	1.741	2.662	0	nc
50.835	-1.089	0.721	2.396	0	nc
49.746	-0.631	-0.077	2.270	0	nc
49.115	-0.360	-1.254	2.206	0	nc
48.756	-0.369	-1.908	2.184	0	nc
48.387	-0.708	1.171	2.139	0	nc
47.679	-1.614	4.213	2.055	0	nc
46.065	-2.277	3.610	2.419	1	nc
43.788	0.463	0.083	2.249	1	C
44.251	2.599	-2.250	2.314	1	C
46.850	1.334	-1.283	2.507	1	C
48.183	0.545	-0.707	2.410	1	C
48.729	0.349	-0.516	2.394	0	C
49.077	0.050	0.045	2.211	0	nc
49.128	-0.162	0.227	2.220	0	nc
48.966	-0.168	0.097	2.174	0	nc

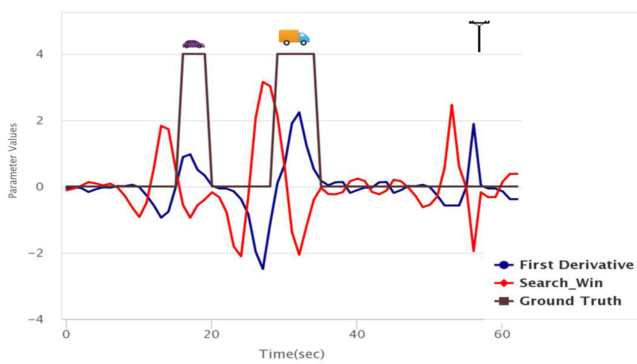


Fig. 8: Both FD and $Search\_Win$ parameters as the user passes by two vehicles and a light pole

the behavior of both FD and $Search\_Win$ parameters as the user passes by both a vehicle and a small truck separated by parking space in one of these experiments. As shown in the figure, not only the base length of the bumps for the truck is wider than the small vehicle, but the base values of the first derivative of the truck is higher than the value of the other vehicle. We plan to further study this in future work to utilize this feature in recognizing vehicle size and type.

We conducted another set of experiments to evaluate whether Magnopark is able to differentiate between parked vehicles and other significant on-street metal objects such as trash bins and light poles. Figure 8 shows the performance of Magnopark for one of these experiments. As shown, MagnoPark detects a pole at second 57. The corresponding detection data for the light pole is shown in Table!II. In this table, it is shown that the pole is detected as one row only (cells are highlighted in red). Since our interesest in this project to detect parked vehicles, MagnoPark is tuned to discard any single or two consecutive rows detected as a vehicle since we assume it is

TABLE II: Calculated features described in Section IV.A for the experiment when the user passes a light pole

Avg	FD	Search_Win	FFT2	GT	Detection
47.975	-0.273	-0.486	3.108	0	nc
47.701	-0.573	0.556	3.036	0	nc
47.128	-0.941	1.828	2.989	0	nc
46.187	-0.759	1.73	2.878	0	nc
45.428	-0.017	0.523	2.754	0	nc
45.411	1.887	-0.558	2.761	0	C
47.298	0.028	-0.176	3.108	0	nc
47.326	-0.06	-0.323	3.108	0	nc

related to other on-street metal objects but not vehicles. As future work, we plan to expand on this observation to explore whether we could locate and map on-street objects other than vehicles such as trash bins and light poles.

2) Realistic/Large-scale Scenarios: Once we evaluated the sensitivity of MagnoPark, we asked one group of users to collect large data sets of magnetometer readings as well as the ground truth data using our application by walking on large number of streets around the campus and at the downtown. As described earlier, users were asked to carry their smartphone in a normal way as they usually carry it and asked to walk as normal as they do (e.g., different walking styles and speeds). Then we used these data to train and build our classifier as described in Section IV.A.

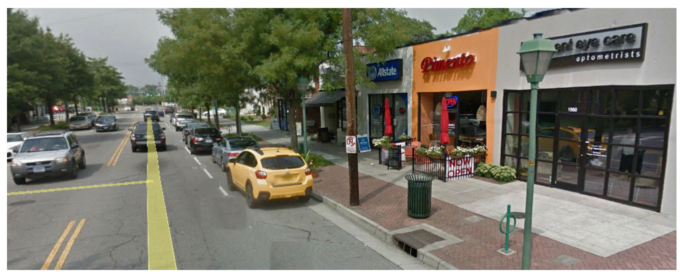


Fig. 9: One of the streets with many cars and parking spots that is used our large-scale experiments

Once we trained and tuned MagnoPark, we conducted several experiments on large sets of streets including the one used in training with another group of users. Note that these experiments were conducted on different times with different traffic loads as well as number of pedestrians. Figure 9 shows one street around the campus of these set of streets. In these experiments, we asked different users to collect multiple data sets corresponding to different sets of streets. We also asked different users to repeat the same set of streets at the same time. For each of these experiments, only one user is asked to collect the ground truth while the others where asked to just walk down the street normally.

To evaluate the performance of MagnoPark under these experiments, we defined the following metrics: *True Positive (TP)* - detecting an actual parked vehicle, *True Negative (TN)* - detecting an actual available parking space, *False Positive (FP)* - identifying a parked vehicle when actually there is no vehicle, *False Negative (FN)* - identifying an available parking spot when actually this spot is occupied by a vehicle, *True Positive Rate (TPR)* - the proportion of occupied parking spots that are correctly identified as such and calculated as TP/(TP+FN), *True Negative Rate (TNR)* - the proportion of available parking spots that are correctly identified as such and calculated as TN/(TN+FP), *Success Rate* - the accuracy of correctly identifying parking as occupied or available and calculated

TABLE III: Configurations of our fours experiment sets as well as the corresponding performance metrics of MagnoPark

Test Sets	Set 1	Set 2	Set 3	Set 4
# Data samples	55300	94700	45230	67503
# Parked Vehicles	56	98	48	67
# Available Parking Spots	74	172	70	95
True Positive (TP)	54	96	45	65
False Positive (FP)	2	1	3	4
False Negative (FN)	2	2	3	2
True Negative (TN)	72	171	67	91
True Positive Rate (TPR)	0.9643	0.9795	0.9375	0.9701
True Negative Rate (TNR)	0.973	0.9942	0.9571	0.9579
Success Rate	0.97	0.99	0.95	0.96
Error rate	0.03	0.01	0.05	0.04

as (TP+TN)/(TP+FN+TN+FP), while *Error Rate* of MagnoPark is calculated as (1 - Success Rate).

We divided the streets into four large sets in which the numbers of the collected data sets are 55300, 94700, 45320, and 67503 respectively. For these sets, the user calculated searching window size ($Search_Win$) is either 3 or 4. In these experiments the numbers of parked vehicles are 56, 98, 48, and 67 while the numbers of parking spaces between these cars are 74, 172, 70, and 95 respectively. For the first set, for example, while we have 56 vehicles, MagnoPark was able to detect 56 vehicles in which there were no parked vehicles in two of these detected vehicles (false positive). Similarly, while we have 72 available parking spots, MagnoPark was able to detect 74 parking spots in which there were parked vehicles in two of these detected spots (false negative). Table III summarizes the configurations of these fours sets as well as the corresponding performance metrics of MagnoPark.

From the table, we see that Magnopark achieves more than 95% accuracy in the sets regardless of the smartphone location/orientation, users walking styles and speeds, and sidewalk conditions. Therefore, we believe that MagnoPark is a low-cost high-accuracy solution for locating on-street parking spots in metropolitan areas.

VI. Discussion and Future Works

In order to realize MagnoPark in real products, there are several challenges that need to be addressed that we are considering in our future work. Among of these challenges is the differentiation between people walking on-street and off-street. As we described earlier, as several previous works, GPS data could be utilized to detect people activities as well as their locations. This information could be utilized by MagnoPark-Ped application to only send data to the cloud when the person is walking on a street with on-street parking spots. Another challenge is the security issue in which to avoid sending fake or false information. Through authentication and encryption mechanisms, these issues could be guaranteed. Another challenge is that magnetometer sensor requires calibration. Fortunately, sensors on smartphones could rank the quality of their readings and detect whether a calibration is needed. We could build on top of this in which MagnoPark-Ped application would notify their user for a calibration and could stop sending any information to the cloud while waiting for calibration. One other important challenge is the how to incentivize the pedestrians to use this application. From our experiments, we found that only data from few persons on every street would be enough to accurately detect parking spots. For streets with free parking, an incentive like coupons from on-street shops could be used. For streets with paid parking, users could get paid based on the amount of data they collect. Note this approach has been utilized in several crowdsourcing applications.

VII. Conclusion

In this paper, we addressed the problem of finding on street parking spaces using the pedestrian sensor enabled cellphone. The model proposed approach was discussed briefly. Our approach is fairly general and does not require any prior assumption or prerequisite for both the driver and pedestrians. We build a classification model for our approach and under different scenarios. Finally, we evaluate the performance of our approach under different real scenarios where we consider different walking speeds with different users in different places. We evaluated the performance of our approach in a numerical study. Results show that MagnoPark in most of the scenarios is able to achieve more than 95% accuracy.

References

[1] Barter, Paul, "Is 30% of traffic actually searching for parking?." http://www.reinventingparking.org/2013/10/is-30-of-traffic-actually-searching-for.html, 2013. Online; accessed Dec 20, 2016.

[2] Addison-Wesley, "ON-Street parking." http://www.worldsensing.com/solutions/mobility/solutions-en/on-street-parking.html, 1993.

[3] "Worldsensing, fastprk: The easiest way to park.," Online; accessed Dec 20, 2016.

[4] "Moscow installs smart parking system SENSIT." http://www.nedapidentification.com/news/news/moscow-installs-sensit.html, 2016. Online; accessed Dec 20, 2016.

[5] Sterling, Greg, "Smartphone Ownership Just Shy Of 80 Percent ." http://marketingland.com/167275-167275, March 2016. Online; accessed Dec 20, 2016.

[6] "Compass primer," Wiki, 2016. Online; accessed Dec 20, 2016.

[7] R. E. Kalman, "A new approach to linear filtering and prediction problems," *JJournal of Basic Engineering*, vol. 82, 1960.

[8] S. Taghvaeeyan and R. Rajamani, "Portable roadside sensors for vehicle counting, classification, and speed measurement.," *IEEE Trans. Intelligent Transportation Systems*, vol. 15, no. 1, pp. 73–83, 2014.

[9] Pala, Zeydin and Inanc, Nihat, "Smart parking applications using RFID technology," in *RFID Eurasia, 2007 1st Annual*, pp. 1 – 3, IEEE, 5-6 Sept. 2007.

[10] A. Ibisch, S. Stumper, H. Altinger, M. Neuhausen, M. Tschentscher, M. Schlipsing, J. Salinen, and A. Knoll, "Towards autonomous driving in a parking garage: Vehicle localization and tracking using environment-embedded lidar sensors.," in *Intelligent Vehicles Symposium*, pp. 829–834, IEEE, 2013.

[11] R. Kmmerle, D. Hhnel, D. Dolgov, S. Thrun, and W. Burgard, "Autonomous driving in a multi-level parking structure.," in *ICRA*, pp. 3395–3400, IEEE, 2009.

[12] V. K. Boda, A. Nasipuri, and I. Howitt, "Design considerations for a wireless sensor network for locating parking spaces.," in *Proceedings 2007 IEEE SoutheastCon*, pp. 329–344, 2007.

[13] G. Pierce and D. Shoup, "Sfpark: Pricing parking by demand," university of california transportation center, working papers, University of California Transportation Center, 2013.

[14] S. Mathur, T. Jin, N. Kasturirangan, J. Chandrasekaran, W. Xue, M. Gruteser, and W. 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, (New York, NY, USA), pp. 123–136, ACM, 2010.

[15] S. H. Collins and A. D. Kuo, "Two independent contributions to step variability during over-ground human walking," *Journal of Applied Physiology*, 2011.

[16] M. Uddin and T. Nadeem, "Spyloc: A light weight localization system for smartphones.," in *SECON*, pp. 72–80, IEEE, 2014.

[17] F. Li and et al., "A reliable and accurate indoor localization method using phone inertial sensors," in *Proceedings of the 2012 ACM Conference on Ubiquitous Computing (UbiComp '12)*, (New York, NY, USA), ACM, 2012.

[18] P. Newson and J. Krumm, "Hidden markov map matching through noise and sparseness," in *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '09, (New York, NY, USA), pp. 336–343, ACM, 2009.