ParkNet: Drive-by Sensing of Road-Side Parking Statistics

Suhas Mathur, Tong Jin, Nikhil Kasturirangan, Janani Chandrashekharan,
Wenzhi Xue, Marco Gruteser, Wade Trappe
WINLAB, Rutgers University, 671 Route 1 South, North Brunswick, NJ, USA
{suhas, tongjin, lihkin, janani, wenzhi, trappe, gruteser}@winlab.rutgers.edu

ABSTRACT

Urban street-parking availability statistics are challenging to obtain in real-time but would greatly benefit society by reducing traffic congestion. In this paper we present the design, implementation and evaluation of ParkNet, a mobile system comprising vehicles that collect parking space occupancy information while driving by. Each ParkNet vehicle is equipped with a GPS receiver and a passenger-side-facing 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. Creating a spot-accurate map of parking availability challenges GPS location accuracy limits. To address this need, we have devised an environmental fingerprinting approach to achieve improved location accuracy. Based on 500 miles of road-side parking data collected over 2 months, we found that parking spot counts are 95% accurate and occupancy maps can achieve over 90% accuracy. Finally, we quantify the amount of sensors needed to provide adequate coverage in a city. Using extensive GPS traces from over 500 San Francisco taxicabs, we show that if ParkNet were deployed in city taxicabs, the resulting mobile sensors would provide adequate coverage and be more cost-effective by an estimated factor of roughly 10-15 when compared to a sensor network with a dedicated sensor at every parking space, as is currently being tested in San Francisco.

Categories and Subject Descriptors

C.3 [**Special-Purpose and Application-Based Systems**]:

General Terms

Algorithms, Design, Experimentaion, Measurement

1. INTRODUCTION

Automotive traffic congestion imposes significant societal costs. One study [1] estimated a loss of $78 billion in 2007 in the form of 4.2 billion lost hours and 2.9 billion gallons of wasted gasoline in the United States alone. Several projects have recently sought to address this issue through the design of mobile systems that collect traffic congestion information to improve route finding and trip planning [2, 3]. Unfortunately, a significant portion of traffic congestion is experienced in downtown areas where it is not always possible to reroute a driver. In these densely populated urban areas, congestion and travel delays are also due to parking. In a recent study [4], researchers found in one small business district of Los Angeles that, over the course of a year, vehicles looking for parking created the equivalent of 38 trips around the world, burning 47,000 gallons of gasoline and producing 730 tons of carbon dioxide. Clearly, addressing the problems associated with parking in downtown areas would have significant societal impact, both economically and ecologically.

Lack of information. One key factor contributing to excess parking vehicle miles is a lack of information about road-side parking availability. While occupancy data for parking garages is relatively straightforward to obtain through entry/exit counters, data is generally unavailable for road-side parking. Detailed parking availability information would allow municipal governments to make better decisions about where to install parking meters and how to set prices. Donald Shoup [5] has argued that road-side parking spots are commonly underpriced compared to parking garages, and that this fiscal consideration greatly exacerbates parking problems. Detailed information would allow travelers to arrive at better decisions on mode of travel or use of parking

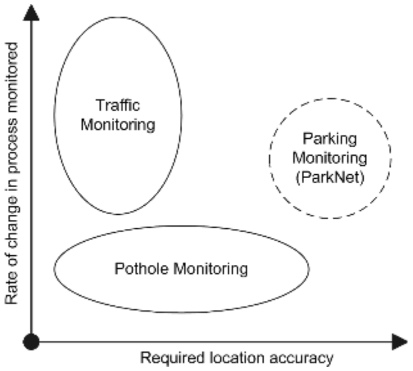

Rate of change in process monitored
Traffic Monitoring
Parking Monitoring (ParkNet)
Pothole Monitoring
Required location accuracy

Figure 1: Categorization of urban sensing applications by required location accuracy and relative dynamics of the process being monitored.

MobiSys'10, June 15–18, 2010, San Francisco, California, USA.
Copyright 2010 ACM 978-1-60558-985-5/10/06 ...$5.00.

garages versus attempting road-side parking. Indeed, several projects are already underway to monitor road-side parking spaces by detecting the presence of parked vehicles over parking spots using *fixed* sensors [6–8]. These efforts rely on sensors installed into the asphalt or in parking meters. This necessitates a large installation cost and operational cost in order to adequately monitor the parking spaces at a city-wide level, or even at the level of a downtown area. For example, the SF-park project [8] aims to cover only 25% of the street parking spots in San Francisco, and will have a cost of several million dollars. Unfortunately, even with such a starting price tag, the cost of such a system does not scale well with the number of parking spaces to be monitored, and is also inherently limited to street-parking with clearly demarcated spaces. A further drawback of such systems is that they require that wireless relay nodes be installed separately on the road side (e.g. in lamp posts) in all areas where sensors are installed in the ground. However, projects such as [8] highlight the magnitude of the problem in large cities and the government's dedication to long-term investments in a smart parking infrastructure.

Drive-by Parking Monitoring. In contrast to such fixed monitoring systems, this paper presents a *mobile* system that collects road-side parking availability information at a lower cost. Our sensing platform consists of a low cost ultrasonic sensor that simply reports the distance to the nearest obstacle and a GPS receiver that notes the corresponding location. Our sensor network leverages the mobility of vehicles that regularly comb a city, such as taxicabs and other government vehicles (parking enforcement, police cars, etc.) to *reduce the number* sensors needed. The cost savings come from the fact that the status of parking spaces in an urban area does not change very rapidly in time, and hence continual sensing through fixed sensors is unnecessary. Realizing this application, however, requires that several unique challenges in mobile systems be overcome that have not been addressed in prior efforts in mobile sensing.

In order to place our work in a broader context, consider the diagram presented in Figure 1, where we have placed our ParkNet system relative to several notable vehicular monitoring efforts in terms of the required location accuracy needed by the sensing application as well as the underlying rate of change of the event being monitored. For example, in [9], a system is presented that monitors the presence of potholes in road surfaces. This involves monitoring a very slowly changing quantity with moderate location precision. Traffic monitoring systems, such as [2] on the other hand monitor a more rapidly changing quantity, but require relatively low precision. In contrast, ParkNet can be considered to require significant spatio-temporal accuracy as the occupancy of parking spaces can vary on the order of minutes and, further, some applications in ParkNet might require significant location accuracy in order to associate cars in specific parking slots.

In our work, we have overcome the underlying challenges of dynamics and location accuracy associated with parking monitoring applications, through the careful integration of ultrasonic measurements with GPS readings that are corrected through environmental fingerprinting. Our ParkNet system has been tested experimentally, collecting over 500 miles of road-side parking data over two months, and our results show that such a system could be fitted into vehicles that frequently roam downtown areas, such as taxicabs, city buses, or parking enforcement vehicles. Further, we note that there is the potential to reuse ultrasonic rangefinders already integrated in some modern vehicles for parking assist and automated parking applications.

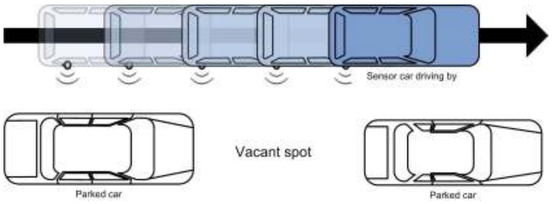


Figure 2: Ultrasonic sensor fitted on the side of a car detects parked cars and vacant spaces.

Overview. In Section 2, we provide an overview of the challenges associated with identifying parking locations and their occupancy. In Section 3, we detail the system that we have built for monitoring parking, providing the rationale for the system choices that we have made. We next explore in Section 4 the ability of our system to monitor parking spaces. Since one potential application we envision involves associating cars with their specific, corresponding parking slot, in Section 5 we next detail an approach to improve location accuracy sufficiently to support such an application. In Section 6 we turn to exploring how many vehicles should be part of such a mobile system in order to adequately monitor parking slots. We summarize the lessons learned in Section 7 and place our work relative to related work in Section 8. Finally, we conclude the paper in Section 9.

In summary, the key contributions of this paper are:

- Demonstrating the feasibility of a mobile sensing approach to road-side parking availability detection through the design, implementation, and evaluation of such a system. Our experimental evaluation uses over one month of data from up to three vehicles passing through the downtown Highland Park, NJ area;

- Proposing and evaluating an improved approach to GPS positioning using *environmental fingerprinting* that allows us to achieve the location accuracies necessary for precise matching of cars with their associated parking slots;

- Showing through trace-based simulations with a dataset involving San Francisco taxis that a few hundred taxis provide adequate spatio-temporal sampling of a downtown area, which is precisely where parking is most scarce.

2. THE ROAD-SIDE PARKING CHALLENGE

Finding street-side parking in a crowded urban area is a problematic task and one that most drivers dread. Finding a parking space near one's destination could be much easier if there were a way to know ahead of time which areas have available parking spaces. Often times, a street only a few blocks away might have vacant parking spaces but a driver looking for parking has no way of knowing this.

One approach to addressing the road-side parking problem may be a spot reservation system that allows vehicles claim available spots before they arrive at their destination. This

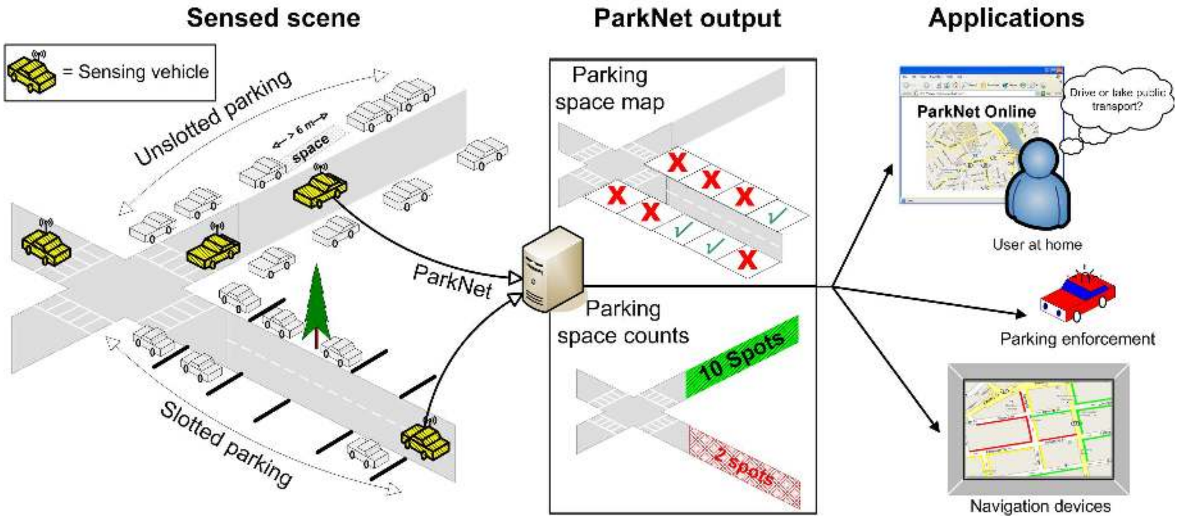


Figure 3: A diagram depicting the various scenarios and events involved in the detection of parking space using mobile sensors.

approach is difficult because it (i) requires exact knowledge of the available road-side parking spots at any given time, (ii) requires all other vehicles to be notified of and to obey reservations, (iii) may lead to inefficiencies if drivers with reservations do change their plans or experience significant travel delays. While this approach presents interesting research challenges, we chose to focus on a different approach that in our opinion has more potential for near-term impact: presenting drivers and municipal governments with near-real-time information and detailed historical parking statistics.

Value of Real-time Information. As Donald Shoup has argued [5] municipalities already posses parking management tools such as parking meters and pay stations and a large share of excess vehicle miles due to the search for parking could be eliminated through basic road-side parking price adjustments. Shoup concludes that prices should be set to achieve an 85% occupancy rate on each block. This approach, however, would require detailed occupancy rate information that allows parking authorities to adjust prices and to determine which city areas should be included in the pricing scheme.

Beyond adjusting road-side parking prices, detailed parking availability statistics could be widely disseminated on web-based maps or navigation systems which would incur the following further benefits:

- Improve traveler decisions, with respect to mode of transportation, the choice of road-side parking vs parking garage, and in which area to search for road-side parking,

- Suggesting parking spaces to users driving on the road looking for parking, through a navigation device or cellphone,

- Allow parking garages to adjust their prices dynamically to respond to the availability or non-availability of parking spaces in the immediate area, and

- Improve efficiency of parking enforcement in systems that utilize single pay stations for multiple parking spaces – parking enforcement vehicles can detect the presence of a parked vehicle in a space that has not been paid for.

Parking information in slotted and unslotted areas. To define concrete parking metrics it is helpful to distinguish areas where vehicles are arranged in slots with demarcated parking bays (often separated by lines marked on the road), which we refer to as slotted areas, from areas without any marked parking spots, which we will call unslotted. Slotted parking space are typically used where parking meters or other parking pay stations are installed. This is arguably the more important case, because parking is usually slotted and metered in the areas where parking is most scarce. In such areas it is easier to measure the number of available parking spaces, because the spacing between cars is regulated. We consider two types of parking information:

Space Count. The number of parking spaces available on one given road segment, which is simply the total number of marked parking spots less the occupied slots.

Occupancy Map. A map showing each parking slot as occupied or vacant. This is more detailed representation of the parking scenario which will be of particular interest in assisting parking enforcement.

We expect that periodic per-block space counts are sufficient for many parking information applications. Occupancy maps are immensely valuable for parking enforcement. A parking enforcement vehicle with a sensor and connectivity to a database that keeps track of which slots have been paid for, would be able to determine whether there is a car parked in a slot that has not been paid for, or whose time has expired. This is particularly relevant for street-parking areas with a single payment machine for a large group of slotted spaces, since the lack of parking meters for individual spaces makes the task of finding offending vehicles harder.

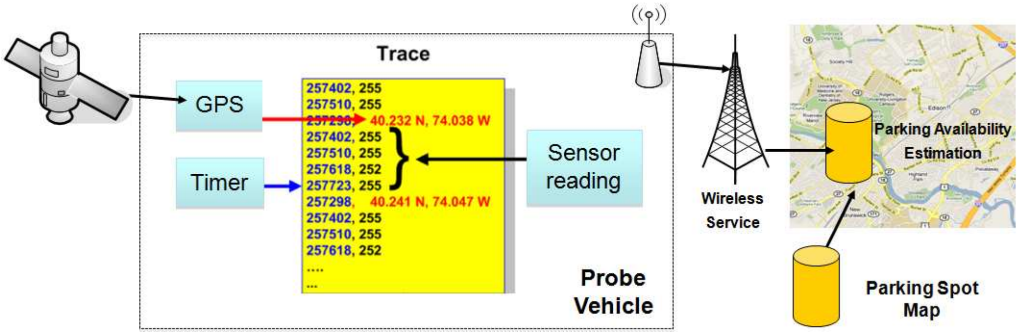


Figure 4: Schematic diagram explaining the overall architecture of the system.

In the case of unslotted street parking, the number of available slots is not defined a priori and depends on the length of vehicles. Still some parts of the road might be marked as no parking zones either explicitly or implicitly by the presence of driveways or fire hydrants for example. To define a space count for unslotted areas, we measure the distances $d_1, d_2, \ldots\ldots d_n$ of all available stretches of valid parking (which are bounded by parked vehicles or no parking zones) on a given road segment. The number of spots is then defined as $n = \sum_i \lfloor d_i/d_{spot} \rfloor$, where d_{spot} is taken to be the (fixed) size of one parking spot (typically $\sim$ 6 meters). The equivalent of the occupancy map for the unslotted model will be the series of available parking stretches $d_1, d_2, \ldots, d_n$, together with the starting latitude/longitude location stamp of each stretch.

Indeed, some municipalities have already recognized the value of such detailed parking statistics and are installing sensing technologies. The city of San Francisco, for example, is presently installing a stationary sensor network to cover 6000 parking spaces under the SFPark project [8]. This network utilizes a sensor node installed in the asphalt in the center of each parking spot. This node detects the presence of a vehicle using a magnetometer among other sensors and forms a mesh network to deliver the data to a centralized parking monitoring system. To ensure connectivity, the mesh network also requires repeaters and forwarding nodes on lamp posts and traffic lights.

Installing a dedicated sensor network for monitoring parking information is relatively expensive, due to the installation and maintenance costs. According to a Department of Transportation report [10], the installation cost of typical per spot parking management systems ranges from \$250-\$800 per spot. While we do not know the exact cost of the system used by the SFPark project, the total project volume including smart parking management functions is 23 million dollars [8]. Furthermore, fixed sensors are quite difficult to place in areas without marked parking slots.

What if it were possible to obtain *most* of the information on the occupancy of parking spaces, at a much lower cost? We believe sensing spaces using a collection of mobile sensors can provide such a solution because turnover on one given parking spot is on the order of tens of minutes in the most expensive downtown areas and hours in many more residential city areas. Thus the required per-spot sampling rate is relatively low and the use of one dedicated sensor per spot appears wasteful. Intuitively, a single mobile sensor can do the work of hundreds of fixed sensors and if we accept the limitations of a probabilistic system, costs can be reduced further by mounting the mobile sensors on existing vehicles that roam the city, albeit perhaps on somewhat unpredictable paths.

2.1 Design Goals and Requirements

We identify the following design goals and requirements that constrain our solution:

Provide Parking Statistics. The drive-by monitoring system should be able to determine the availability of road-side parking spaces on at least an hourly basis with sufficient accuracy to (i) direct drivers to areas with several available parking spots and to (ii) inform municipal government parking management decisions.

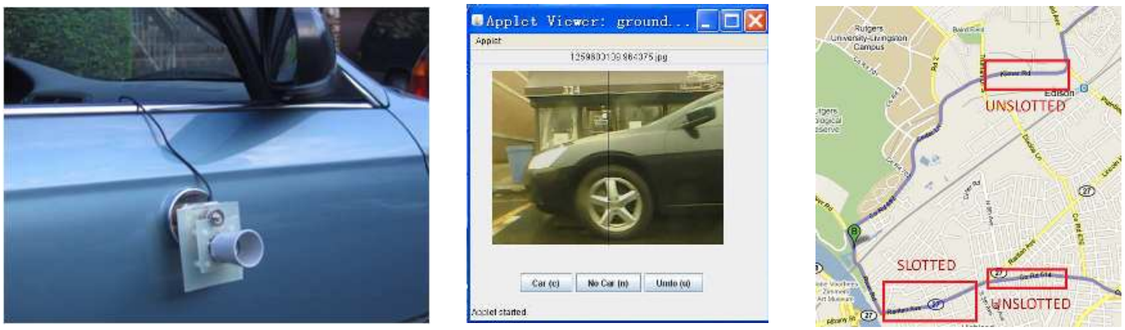
Assist Parking Enforcement. Given a map of paid-parking spaces the drive-by sensing system should be able to identify candidate parking spots occupied by an infringing vehicle. Accuracy should be sufficient to assign human parking enforcement personnel. The system is not intended to generate automatic citations.

Low-cost sensors. The system should operate with sensors that are typically used in automobiles for other applications. This rules out more expensive specialized sensors such as laser scanners.

Low vehicle participation rates. While one could envision that eventually all vehicles simply report their parking locations as obtained from the Global Positioning System, this would require the participation of nearly every vehicle to achieve high data accuracy. Given vehicle lifetimes of 10+ years in the United States, full deployment is difficult to achieve without government regulation mandating installation in every new vehicle or retrofitting of vehicles.

3. DRIVE-BY SENSING OF PARKING AVAILABILITY

The ParkNet architecture employs a mobile sensing approach with ultrasonic rangefinders and GPS to monitor road-side parking availability. It also introduces an environmental positioning concept to achieve the positioning accuracy necessary to match vehicles to demarcated parking



(a) (b) (c)

Figure 5: (a) An image of the ultrasonic sensor side-mounted on a car (b) The java applet we used for recording ground truth from images. (c) The map of the data collection area.

slots. As illustrated in Fig. 4, several sensors-equipped vehicles report their sensor readings to centralized parking estimation server. This server combines information from a parking spot map, which may be available in different levels of detail (as we will discuss in Sec. 4 with the sensor readings from one or multiple vehicles obtained on the same road segment to create an estimate of road-side parking availability. Vehicles can report their data over a cellular uplink but opportunistic use of Wifi connections is also possible depending on cost/delay tradeoffs. The parking availability information can then be distributed to navigation systems or distributed over the Internet, similarly to the various distribution channels for road traffic congestion information.

3.1 Choice of Ultrasonic Sensors

We chose ultrasonic rangefinders because of their relatively low cost of tens of dollars compared to laser rangefinders and automotive radars, better nighttime operation compared to cameras, and their increasing availability in cars to support parking assistance and automated parking functions in modern vehicles. This potentially allows reusing already present sensors in future vehicles.

Each sensor vehicle in our set-up carries a passenger-side facing ultrasonic rangefinder to detect the presence or absence of parked vehicles. It's range should be equal to at least half the width of urban roads and the sampling rate high enough to provide several samples over the length of a car at maximum city speeds. Figure 5(a) depicts our prototype incarnation using a Maxbotix WR1 waterproof rangefinder, magnet-mounted to the side of a vehicle. This sensor emits sound waves every 50 ms at a frequency of 42 KHz. The sensor provides a single range reading from 12 to 255 inches every cycle, which corresponds to the distance to the nearest obstacle or the maximum range of 255 if no obstacle is detected. The sensor measurements at each vehicle are time-stamped and location-stamped with inputs from a 5Hz GPS receiver, producing the following sensor records:

```
<Kernel-time, range, latitude, longitude, speed>
```

Vehicles transmit a collection of these measurements to the parking estimation server where data from mobile sensors is continuously aggregated and processed using probabilistic detection algorithms that we will describe in the following sections.

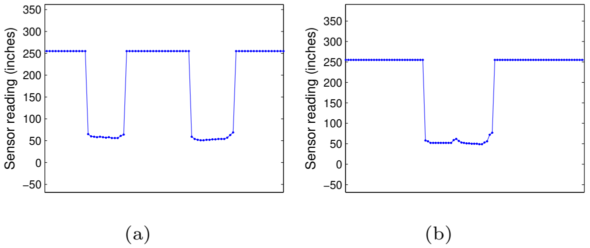
3.2 System Specifications

The on-board PC has a 1GHz CPU with 512 MB RAM, 20 GB hard disk space, an Atheros 802.11 a/b/g mini PCI card, and 6 USB 2.0 ports. We used a Garmin 18-5Hz GPS with 12 channel receiver that provides 5 fresh GPS readings per second, and a real-time WAAS correction of errors less than 3 meters. Both the sensor and the GPS provide data in serial format, which can be accessed via an USB serial port on a computer. Note that while we use an off-the-shelf GPS unit, in practice, there exists the opportunity to use built in GPS units in vehicles, which are sometimes wired together with the vehicle's odometry to allow for better location accuracy.

3.3 Prototype Deployment

We implemented and deployed this system on three vehicles which collected parking data over a 2 month time frame during their daily commute. Specifically, data was collected in three road-side parking areas in Highland Park, New Jersey as depicted in Figure 5(b). One of these areas contained 57 marked slotted parking spots. The two unslotted areas are 734 m and 616 m in length. During the experiment time frame, we collected a total of more than $\sim$ 500 miles of data on streets with parking. All data collected was from roads with single lanes (see Section 7 for a discussion on multi-lane roads). The data collection was not controlled in any manner (e.g. speed, traffic conditions, obstacles, etc.) – all data was collected while drivers went about their daily commutes at various times of the day, oblivious to the data collection process.

To obtain ground truth information for system evaluation purposes and to be able to analyze erroneous readings, we integrated a Sony PS3 Eye webcam into the passenger-side sensor mount. To avoid angular and shift errors with respect to the sensor, we mounted the camera just above the sensor and aligned its orientation to the sensor. A user space program captures about 20fps and tags each image with a kernel time stamp. This time stamp links images to the sensor records obtained at approximately the same time. We then manually inspected each image and entered the ground



(a)

(b)

Figure 6: Dips in the sensor reading as a sensing vehicle drives past (a) two parked cars with some space between them, and (b) two very closely spaced parked cars

truth sensor data. For this process we implemented the java applet depicted in Fig. 5(b). It displays each image together with a reference line marking the estimated aiming of the ultrasound sensor and allows the human evaluator to enter whether the reference line crosses a parked vehicle. Note that the webcam is not part of the proposed parking sensing system, it is purely used for evaluation and data analysis purposes.

3.4 GPS trip-boxes for limiting data collection

We limit data collection to the parking regions identified in Fig. 5(c) due to the relatively small areas of roadside parking on one commute trip and the large volume of video data involved. The activation and deactivation of data collection is implemented in our system by using the idea of a tripbox. Tripboxes are derived from our virtual trip line concept [3], but represent rectangular areas defined by two (latitude,longitude) points. Each tripbox is also associate with an entry and an exit function, which starts and stops data collection, respectively. The tripbox daemon simply reads the current GPS coordinates from the GPS receiver and checks whether it falls inside or outside the tripbox region. If the current coordinate is the first instance of the mobile node inside the tripbox, it triggers the entry function. In case the mobile node is already inside the tripbox and the next received coordinate is outside this region, it triggers the exit function. We use tripboxes because it simplifies the handling of vehicle routes, which might enter a parking zone from an unexpected direction, or the acquisition of a GPS fix while already inside a trip box.

Since GPS coordinates can oscillate due to positioning errors, the tripbox implementation includes a guard distance and a guard time to avoid repeatedly triggering the same tripbox functions. The guard distance is a minimum distance that must be traveled in between two tripbox boundary crossings. Similarly, the guard time is the minimum time that must be spent before the next tripbox function can be triggered. This avoids triggering the start and the stop functions repeatedly due to GPS errors.

4. DETECTION OF PARKING SPACES

The detection algorithm translates the ultrasound distance-reading trace into a count of available parking spaces. The distance-reading trace provides a one-dimensional view of

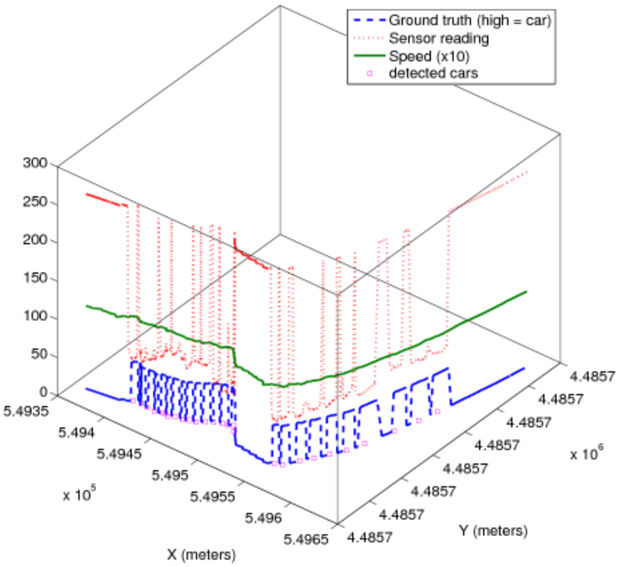


Figure 7: An example plot showing the sensor reading (dotted red) and ground truth (dashed blue line, high = car, low = no car), speed (increased in magnitude by x10 for visual clarity), and the output of the detection algorithm (purple squares).

the distance to the nearest obstacle as the sensing vehicle moves forward. Figure 6 (a) shows an example of the trace produced by our sensor as a sensing vehicle drives past two parked cars. We will refer to the features in Figure 6(a) as *dips* in the sensor reading. The width of a dip is representative of the length of a parked car, although, as we shall see, the errors in location estimates obtained from a GPS receiver can distort the true length of the car in a somewhat random manner. We assume that maps of areas with street-parking slots are available from another source (discussed further in Section 7).

4.1 Challenges

An ultrasonic sensor does not have a perfectly narrow beam-width, but instead the beam width of the sound waves emitted widens with distance. This implies that the sensor receives echos not just from objects that are directly in front but also from objects that are at an angle. This affects how our sensor perceives vehicles that are parked very close to each another. Instead of clearly sensing the gap between these vehicles, the 'dips' in the sensor reading become merged, as depicted in Figure 6(b). Still, classification of the spatial width of the dip allows us to determine the number of cars that a dip corresponds to.

The inaccuracy of latitude and longitude values obtained from the GPS unit adds another challenge to the detection problem. The location estimate provided by a commercial grade GPS receiver suffers from well known errors. Without a priori knowledge of how the GPS error varies in space and time, it is possible that GPS errors can make a parked car appear to be shorter or longer than its true length. Since the detection of parked vehicles depends upon distinguishing objects that are about the length of a car, from other, smaller obstacles in the sensors path (such as trees, recycle

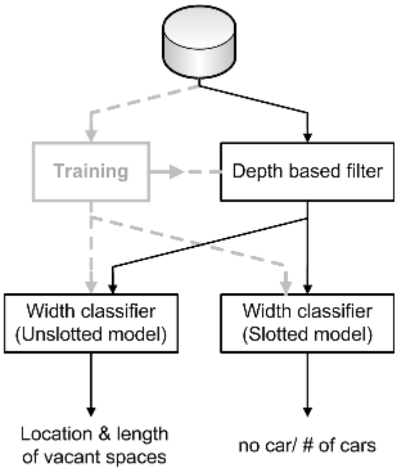

Training
Depth based filter
Width classifier (Unslotted model)
Width classifier (Slotted model)
Location & length of vacant spaces
no car/ # of cars

Figure 8: A series of filters are applied to remove dips in sensor readings that are not caused by parked cars. We use 20% of our data to train our model and the remaining 80% to evaluate its performance.

bins, people, etc.), this sometimes leads to false alarms (i.e. dips caused by objects other than cars to be classified as parked cars), and missed detections (i.e. parked vehicles to be classified as something other than a parked car).

4.2 Detection algorithms

Slotted model. Each dip in the sensor trace has depth and a width, that correspond to the distance from the sensor of the object causing the dip, and the size of the object in the direction of motion of the sensing vehicle. The sensor trace is first pre-processed to remove all dips that have too few readings (less than 6 sensor readings, assuming a maximum speed of 37 mph and a car length of 5 meters) and could not possible have arisen from a parked car. To detect a parked car, the width and depth of each dip in the sensor reading is compared against thresholds. We determine these thresholds using part of our data for training the system. Figure 8 shows a series of *filtering* stages that are applied to each detected dip in the sensor reading. Figure 9, shows the depth and width of the peaks observed in 19 separate trips in an area with slotted parking. We used this data for jointly picking thresholds for the depth and width of a sensor-reading dip that provide the minimum overall error rate (i.e. the sum of the false positive rate and the miss detection rate[1]). These thresholds were determined to be 89.7 inches for the depth and 2.52 meters for the width, resulting in an overall error rate of 12.4%.

Finally, all remaining dips are checked for spatial width, and compared against a threshold representing the typical length of a car. For this, we convert the interpolated GPS coordinates belonging to the starting and ending sample of the dip to UTM (meters) and compute the distance in meters between the starting and ending sample. Since some dips correspond to multiple cars parked very close together, we classify dips of a width greater than twice the threshold for one car, to belong to two cars, and so on. This allows us to

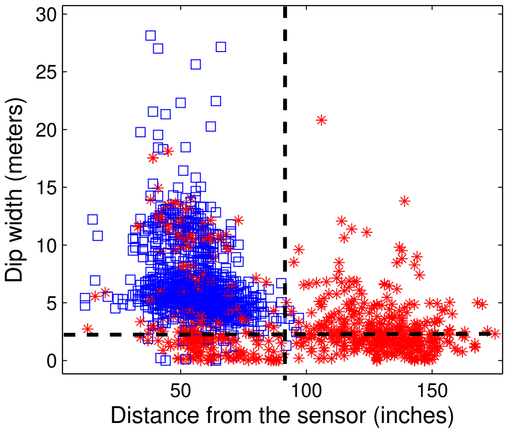


Figure 9: A plot of the depth and width of most prominent dips observed in the sensor reading, caused by parked cars (blue squares) and objects other than cars (red stars). This data set is taken from 19 trips in an area with slotted parking and is used for training the model used for classifying the rest of the data.

count the number of cars on a stretch of road. Subtracting this from the total number of slots on the road, as given by the map, provides an estimate of the number of vacant spaces.

Unslotted model. For the unslotted parking model, the number of cars that can be accommodated on a given stretch of road depends upon the manner in which cars are parked on it at any given instant of time. Since each successive pair of parked cars in this model can have a variable amount of space between them, we must estimate the space between successive parked cars to determine whether the space is large enough to accommodate one or more cars. To accomplish this, we use the sensor trace to estimate the the spatial distance between dips that have been classified as parked cars. The estimated length of the vacant stretch is then compared against the length of a standard parking space (which we have taken to be 6 meters).

4.3 Metrics

We will deal with the slotted and unslotted street-parking models separately and will assume that it is easy to obtain information about which streets have which type of parking as prior knowledge. For the slotted model, we are interested in detecting how many of the parking spaces on a road segment are vacant.

Let us assume that a street segment with the slotted parking model is known to have N parking slots and that at a given instant of time, n of these slots are vacant. A sensing vehicle that drives through this street determines that $\hat{n}$ of the slots are vacant. The value of $\hat{n}$ can differ from n due to missed detections as well as false positives. We are interested in the missed detection rate p_m, i.e. the probability that a parked car is not detected, and the false positive rate p_f, i.e. the probability that there is no parked car in a given slot but the detection algorithm detects one.

The ratio $\hat{n}/n$ captures the performance of the detection

[1] The overall error rate is minimized when the false positive rate and the miss detection rate are equal in value [11].

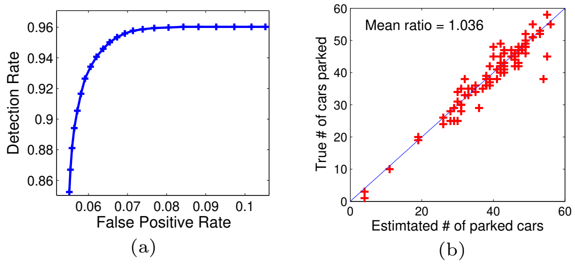


Figure 10: (a) Detection rate versus false positive rate for the slotted parking model. (b) A scatter plot showing the number of vehicles detected against the actual number of vehicles parked for the slotted parking model. Each data point represents a separate run

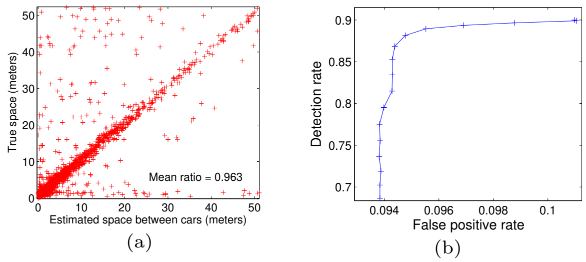


Figure 11: (a) Scatter plot showing the estimate of the space between cars Vs. the true space as obtained from video measurements. (b) Detection rate versus false positive rate for the unslotted parking model, assuming at least 6 meters for a car to park.

algorithm in estimating the number of vacant spaces. This ratio can be smaller or larger than 1, for a given run, depending on whether there are greater number of missed detections or false positives. Since our thresholds for dip classification are chosen from our training data to minimize the overall error rate, and this is known to occur when the probability of false alarm equals the missed detection probability [11], we expect that the ratio $\hat{n}/n$ to have a mean close to 1.

For the unslotted model, the appropriate metric of interest is: 'How many more cars can be accommodated on a given road segment, given the cars that are presently parked on it?'. As explained in Section 4.2, estimating this number requires estimation of the space between parked cars. As in the slotted parking model, we will assume that we have available to us, the locations of stretches where unslotted parking spaces are available and we will run our detection algorithm only over such stretches. Whenever the detection algorithm ascertains that a space between two parked cars is large enough to accommodate another car, it records the estimated space $\hat{d}$. Suppose the actual space between the cars is d, then $\hat{d}$ can be larger or smaller than d and as before, we will take the measure of accuracy to be $\hat{d}/d$. Further, we are interested in the miss detection rate p_m, i.e. the probability that our algorithm decides that there isn't enough space for a single car, when there actually is, and the false positive rate p_f, i.e. the probability that the detection algorithm declares that one or more cars can be accommodated in a space between two parked cars, whereas in reality there is not enough space for a single car. In our evaluation, we will assume a vehicle of length 5 meters and at least half meter on either side for parking, for a minimum of 6 meters to qualify for a parking space.

4.4 Evaluation

To evaluate our detection algorithm, we utilize the images recorded by the webcam in our set-up. Since the camera records images at a rate of 21 frames per second, it matches the rate at which sensor readings are recorded fairly well. Each image is manually labelled based on whether the center of the image has a car in front or not. The time stamp associated with each image allows us to interpolate a location stamp for each image. This provides the ground truth for both our training data-set and the evaluation data-set. Figure 7 shows an example of a typical trace of the sensor reading along with the ground truth. Also shown in the image are the speed of the car and the cars detected as output of our detection algorithm. Figure 10(a) shows the tradeoff between detection rate and false positives for the slotted model, as the threshold for the width of a dip (i.e. corresponding to the length of a car) is varied. We found that a threshold of 2.5 meters provides the best tradeoff in the minimum probability of error sense. Figure 10(b) shows the number of detected parked vehicles on a road with 57 parking slots, against the true number of parked cars. We found that on average, the ratio of the estimated number of cars to the true number of cars is 1.036, indicating a fairly good estimator of the availability of free spaces.

For the unslotted model, we compare our estimate of space between two successive cars with the true value as computed using the ground truth generated by our tagged video images. The plot in Figure 11(a) shows this comparison as a scatterplot. The estimates space is on average 96% of the true space. Further, the estimated space is compared with the length of a typical parking slot (usually about 6 meters) to determine whether an additional car can be accommodated. The result of this detection leads to false positives and missed opportunities, and the trade-off between the corresponding false positive rate and missed detection rate is shown in Figure 11(b), as the threshold for the width of a dip is varied. Figure 12 shows two examples of cases, as captured by our webcam, where the detection algorithm was fooled into making false alarm decisions.

Given that our estimate for the number of cars parked in the slotted model and the amount of space between successive cars are 1.036 times and 0.963 times the true number of cars and the true space respectively, we can say that the system is approximately 95% accurate in terms of obtaining parking counts. In the following section, we will address the problem of trying to assign detected cars to specific slots in the slotted model.

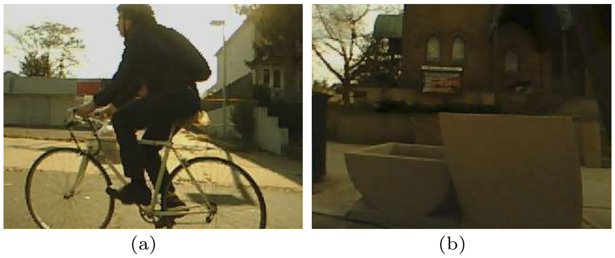


Figure 12: (a) A moving bicyclist, and (b) a flowerpot, both objects that produced dips in the sensor trace that were classified incorrectly as parked cars

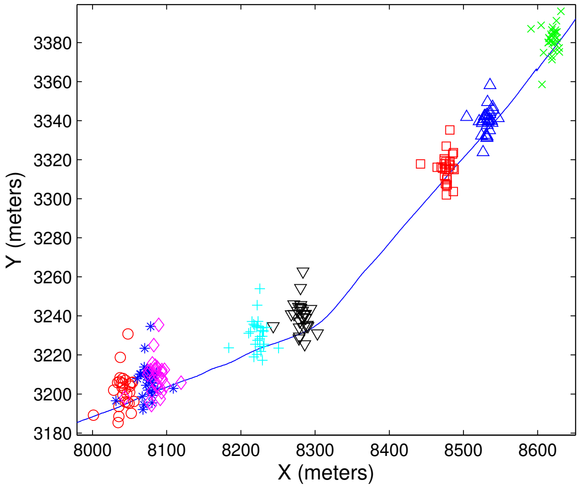


Figure 13: The locations of 8 objects along a road shown for 29 different runs.

5. OCCUPANCY MAP CREATION WITH ENVIRONMENTAL POSITIONING CORRECTION

While counting of available parking spaces did not require high absolute position accuracy, creating an occupancy map of parking increases accuracy requirements since a detected car has to be matched to a spot on a reference map. The location coordinates provided by a GPS receiver are only typically accurate to 3m (standard deviation) when the Wide Area Augmentation System (WAAS) service is available [12, 13]. Given a parking spot length of about 7m, one can expect a significant rate of errors—any error greater than 3.5m could lead to matching a vehicle to an incorrect adjacent spot.

To address the occupancy map challenge, we develop a occupancy map creation algorithm that exploits both patterns in the sequence of parking spots as well as an Environmental GPS position correction method to improve location accuracy with respect to the parking spot map. We first study how the error in GPS coordinates behaves as a function of distance. The positioning accuracy of a GPS receiver is affected by several factors including ionospheric effects, satellite orbit shifts, clock errors, and multipath. Ionospheric effect typically dominate the other error sources, except for errors that experience satellite occlusion (e.g., in urban canyons). Ionospheric effects remain similar over distances of several 10s of kilometers and they contain significant components whose rate of change is on the order of ∼10s of minutes or longer. GPS errors can therefore be expected to be correlated in time and space. However, the Wide Area Augmentation System was designed to reduce these ionospheric and some other errors, raising the question whether the resulting GPS errors with WAAS still exhibit strong spatio-temporal correlation.

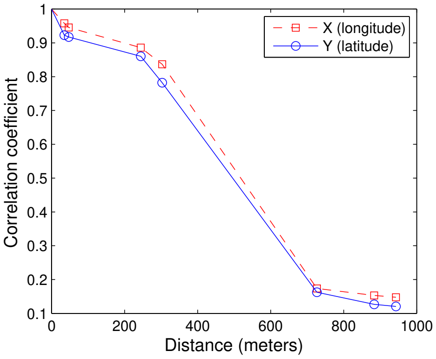


Figure 14: Correlation in the error (with respect to centroid) of the GPS location estimates corresponding to fixed points along a road, as a function of distance along the road.

We find that the GPS error is in fact highly correlated at short distances, and the correlation tapers off with distance. Motivated by this observation, we propose a method to improve absolute location precision by an environmental fingerprinting approach. In particular, we use the sensor reading to detect certain fixed objects that persistently appear in our ultrasound sensor traces, and utilize these to correct the error in the GPS trace. To validate the approach, we test it on the slot-matching problem described above. We expect that our environmental fingerprinting approach will benefit any mobile sensing application that requires precise estimates of location or distance between two points, as is the case in some of the scenarios in our sensing application.

5.1 GPS Error Correlation

We began our study by location-tagging certain fixed objects (such as trees, recycle bins, the edges of street signs, etc. which would also be picked up by our sensor) in our video traces on a given street over multiple different runs from different days. We tagged the data with the same video tagging application we developed for evaluating our detection algorithm (see Figure 5(c)). We found, as expected that the tagged coordinates for a given object from multiple runs varied significantly. Using 29 different runs and 8 objects on a street, we found the standard deviation of error to be 4.6 m in the X-direction and 5.2 meters in the Y-direction. We note here that the error due to variation in the lateral position of the sensing vehicle was not corrected for because the street chosen for this was narrow enough to allow the lateral variation to be within ±1/2 meter. Also this street

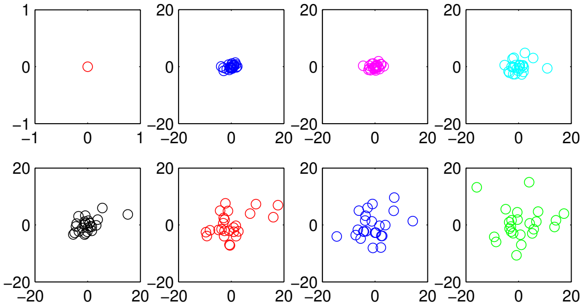


Figure 15: Using the first object in a series of 8 objects to correct the error of the remaining 7 objects. The plots are arranged in increasing order of distance from object 1, from left to right and top to bottom. The colors match those of objects in Figure 13 (axes are in m). Location error builds up with increasing distance from the object 1.

was almost parallel to the X axis and so we expected to observe an larger error in the Y direction to slight variaions in the sensing vehicle's lateral position.

We also found that the error between GPS coordinates is correlated from one object to the next. Figure 13 shows the locations of the 8 objects along the street. We chose the centroid of the 29 tagged locations for each object as the reference location and subtracted each tagged location coordinate to compute the error. Figure 14 shows the correlation between the error in the X and Y directions as a function of distance along the street, using the 8 objects we selected. It is worth noting that the correlation in the error is fairly high for a distance of up to ∼ 250 meters.

5.2 Environmental Fingerprinting

The above investigation suggests that if the GPS error is corrected at a given point, then it is likely to remain corrected for an appreciable distance. In Figure 15, we utilize the location-stamp of the first object on the street (lower left corner in Figure 13) to correct the errors in the location of the remaining 7 objects. As Figure 15 illustrates, the residual error in the error-corrected location-stamp for the 7 objects increases with increasing distance from object 1.

Fingerprinting the environment by relying on features in the sensor trace that are produced by fixed objects in the environment, provides a possible means to improve location accuracy beyond that provided by GPS alone. However, fingerprinting a street requires multiple traces from that street, from which the locations of objects that are very likely fixed can be determined.

Estimating the GPS error using the sensor trace involves a simple task — comparing the reported location of the pattern (dips) produced by a series of fixed objects to the a priori known location of this pattern (as determined from multiple previous traces from the same road segment). The offset between the two gives an estimate of the error in the reported location.

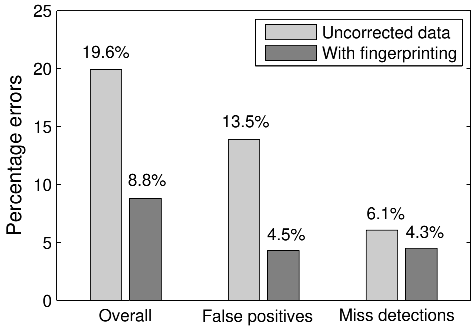


Figure 16: A comparison of the error rates in assigning parked cars to the correct slots with and without error correction using fingerprinting.

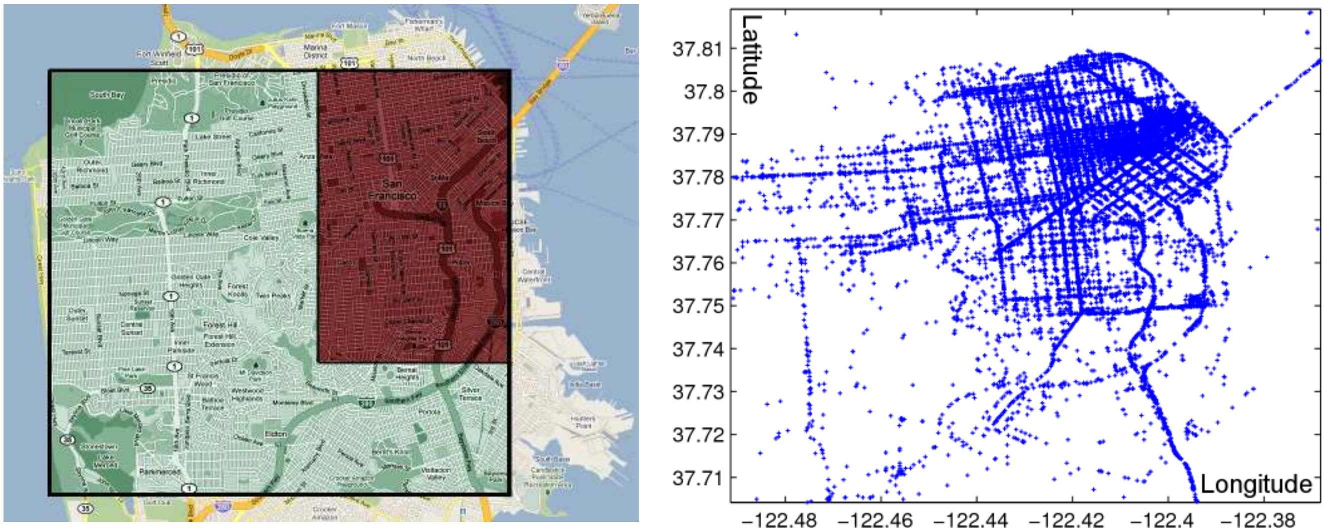
For example, to detect the dips corresponding to two successive fixed objects from an experimental trace, we first identify a set of candidate dips for each object from the dips that are *not* classified as vehicles — each candidate set consists of dips within a radius of 20 meters of the known mean location of the fixed object (mean computed from past traces). We then select one dip from each candidate set so that the distance between the successive selected dips best matches the known distance between the mean locations of the objects to which they correspond. The vector offset between the known locations and the reported locations of the objects is the GPS error estimate. The correction procedure must be repeated with another set of objects once the vehicle travel distance has exceeded the correlation distance.

For n such objects, $i = 1, \ldots n$, we recorded the location stamps $l_i(x, y)$ of the dips corresponding to each object and subtracted it from the known true location of the object $t_i(x, y)$ (assuming the centroid of the 29 locations as above), giving an estimate of the error vector $e_i(x, y) = t_i(x, y) - l_i(x, y)$. Next, this error vector from a given object is added to the location estimates of all detected cars that are detected to be within 100 meters of this object.

5.3 Slot-matching

Motivated by our observation of correlation between GPS error in space, we studied the specific application of matching detected parked cars to their respective slots on a street with slotted parking. To accomplish this, the output of the algorithm for detecting cars in the slotted model (see Figure 8) was augmented with the estimated location of each detected car. The locations of 57 slots on a street were determined using a satellite picture from Google Earth. The matching of cars to slots is an instance of the assignment problem [14] on bipartite graphs and can be solved efficiently using the Hungarian algorithm [15]. The problem involves assigning each detected parked car with specified location coordinates in the set of detected cars, to a valid slot from among the set of 57 slots available. The criterion for the assignment is the minimization of the cumulative distance between each car and its assigned slot. We used a MATLAB implementation of the Hungarian algorithm to solve for the slot-matching of detected parked cars.

Figure 16 shows the error-performance of the slot-matching algorithm, when using plain uncorrected traces and with traces that have been corrected using the fingerprinting algorithm described above. We find that the fingerprinting approach described in the previous section segnificantly lowers the error rate in slot assignments.



(a)

(b)

Figure 17: (a) Two areas of San Francisco (one larger and the other, a smaller downtown area) in which we monitored the movements of 536 taxicabs over a stretch of one month (b) Location trace of a single taxicab in San Francisco area over a span of 30 days. The areas with highest presence are also the busiest areas with most street-parking.

6. MOBILITY STUDY

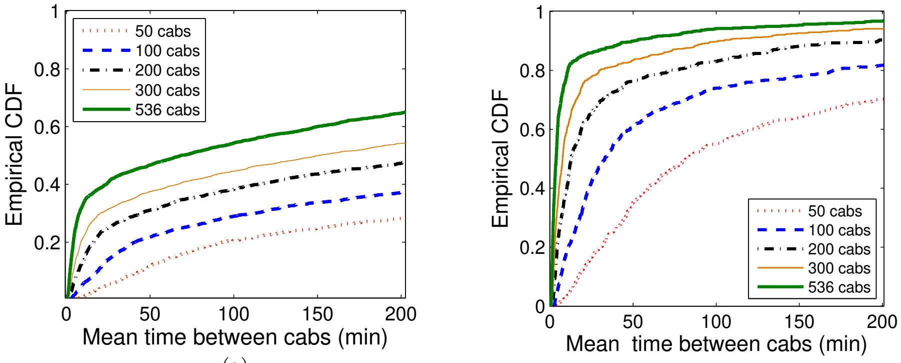
The viability of ParkNet and its desirability over static sensor systems that monitor parking is intimately tied to the number of mobile sensors that must be deployed in order to adequately monitor street-parking spaces. In particular, it is important to determine how often, statistically speaking, a ParkNet mobile node would pass by on a randomly picked street, and how this quantity varies as the number of mobile nodes in ParkNet is varied. This relationship could them be used to determine whether the underlying tradeoff allows for significantly fewer (mobile) sensors to be employed in ParkNet than a system with stationary sensors, and if so, what level of cost savings can be expected.

In order to realistically explore this question, we conducted a study of the mobility patterns of taxicabs using a public dataset of 536 taxicabs in San Francisco collected over a span of roughly one month during 2008 [16]. The dataset contains time-stamped location traces for each taxicab with successive location updates being recorded 60 seconds apart. Figure 17(b) shows the locations of a single taxicab in this data-set over a span of 30 days. We approximated the intermediate locations of each taxicab by linearly interpolating locations between successive GPS updates.

We considered two geographical areas, shown in Figure 17(a): (i) the greater San Francisco area, and (ii) the busiest portion of San Francisco where the business districts and tourist attractions are concentrated. The latter is also happens to contain all the installations of the SFpark project [8]. We conjecture that areas with a greater amount of street parking utilization are also the areas with a greatest presence of taxicabs since both are driven by large concentrations of people. This hypothesis is supported by the observation that all present pilot installations of fixed parking sensors under the SFpark project are in the smaller area in Figure 17(a) (see [8] for a map) and that the taxicab trace also reveals that cabs spend most time in this area (see Figure 17(b)).

We divided each area into a grid of cells that were 175×190 meters in size, and computed for each cell, the mean time between successive visits by a taxicab in the fleet. We chose the size of a cell so that is most cases, only a single road segment is contained within a cell. Our findings are summarized in Figure 18(a) for the larger area and in Figure 18(b) for the smaller, busier area. We find, even with roughly 500 cabs deployed in the greater San Francisco area, the mean time between visits to a cell is on the order of hundreds of minutes. On the other hand, the sampling provided by these same cabs in the smaller, downtown area of San Francisco more than adequately covers the smaller area, with 80% of the cells visited on average with an inter-visit interval of under 10 minutes with just 536 cabs. Using this trace as a guideline for other urban areas, one can extrapolate to estimate the number of taxicabs that must be fitted with a sensor in order to provide a sufficiently small inter-visit time between successive visits to a randomly chosen street.

Cost analysis: A rough analysis of our system reveals that the basic components in our system cost approximately $\sim$ \$400 for one sensing vehicle (a light-weight PC platform: \$170, sensor: \$20, GPS unit: \$100, and \$100 for wiring and connecting components including labor). In comparison, [10] estimates a cost of \$250-\$800 per space for a 'smart parking system'. Even taking a conservative estimate of \$250 per spot, a system consisting of fixed sensors, covering 6000 parking spaces in the upper left corner of San Francisco (as



(a)

(b)

Figure 18: The cumulative distribution function of the mean time between successive taxicabs visiting a street in (a) greater San Francisco (b) smaller, busier downtown area, as the number of cabs being considered part of ParkNet is varied. To compute the CDF, the city is divided into cell of size 175 m × 190 m

a pilot, the SFpark project has installed 6000 sensors in this area of San Francisco at the time of this writing), this would incur a cost of \$1.5M for the fixed sensor system. On the other hand, with only 300 cabs (see Figure 18(b)), which provide an average inter-polling time of $\sim$ 25 minutes for 80% of the cells, the corresponding cost is roughly \$120,000, giving a equipment cost saving factor of 12.5. Note that this number represents a conservative estimate of the cost savings, since (i) in practice, the cost of the mobile sensing system per mobile node can be brought down below our estimate of \$400 with aggressive engineering and mass production, and (ii) we have chosen the lower end of the cost estimate per spot for the fixed sensor system. Note further, that San Francisco city is estimated to have $281,364$ road-side parking spaces, of which $24,464$ are estimated to be metered spaces [17]. While only 6000 spots considered 'high-value' spots by the city of San Francisco have been fitted with fixed sensors, we do not know how many of the total $24,464$ metered spots are within the downtown area we have considered. The cost saving factor is therefore likely to be larger when the actual number of spots in this area is considered.

The operational running cost of each system comprises maintenance and communications costs. We expect the total operational cost of ParkNet to be less than that of a fixed sensor system. Communications in ParkNet can be done over opportunistic WiFi connections in urban environments, making it almost free, and maintenance is expected to be more easily manageable than a fixed system because the number of mobile nodes is much smaller than the number of fixed sensors, and taxicabs are regularly taken in for maintenance into garages.

We note however, that the cost benefit that mobile sensing provides comes from the fact that it provides a non-guaranteed, random sampling of the parking process, whereas fixed sensor provide continuous monitoring wherever they are installed. Hence, the cost associated with our system does increase as we require that a higher fraction of cells be covered (for e.g. extrapolation shows that requiring 95% of the cells be sampled with a mean inter-sampling period of 25 minutes would necessitate that our system be deployed in roughly 2000 cabs). Even so, the cost benefits associated with ParkNet are still significant as it is not necessary to have continual sampling of parking space occupancy.

7. DISCUSSION AND LESSONS LEARNED

Apart from the challenges faced while meeting real-time localization of vacant parking spots, the gathering and processing of reliable sensor data poses its own set of difficulties. We discuss the details of the most challenging issues we encountered below.

Power source. One unexpected difficulty turned out to be noisy sensor data affected by the power source for the in-car nodes. Initially, we used a power inverter to convert the 12 volt DC vehicle power supply to AC power suitable for a standard PC power supply. Although this setup supplied adequate power, it lead to very noisy sensor data. The ultrasound transducer may have been affected by the modified sine wave output of the inverter, which is known to affect sensitive electronic equipment. To address this issue, we installed DC to DC power supplies in each car node and connected them directly to the fuse box. This solution worked in some vehicles, but in vehicles with weaker battery, turn-on of the vehicle node was unreliable.

Multilane roads. Detecting parked cars on a multi-lane road requires lane-detection so that vehicles being passed or passing the sensing car in a different lane are not classified as parked cars. GPS accuracy is unlikely to be sufficient for lane detection. The data collected and reported upon in the paper was taken entirely on single-lane roads. We found, however, through preliminary trials that it is often possible to distinguish moving vehicles in the neighboring lane from parked vehicles by the length of sensor dips. A car moving at similar speeds as the sensing vehicle, for example, will generate a very long dip. Another promising approach is to use a sensor with a much larger range – this can greatly help lane detection.

Speed limitations. A limitation of using an ultrasonic sensor is that we are limited by the speed of sound. Our sensor provides a maximum range of 6.45 meters, and since it must wait for a return echo before sending out the next pulse, the sensor provide only 20 distance readings per second. This implies that if the sensing vehicle is driving too fast, it will not be able to sense a parked vehicle. For example, at a speed of 15 meters per second (roughly 33 miles per hour), a 5 meter long parked car should produce about ~ 6 distance readings from the sensor. However, the speed limit in areas with street parking is usually in the range of 35– 40 miles per hour and so our choice of sensor should not be a limiting problem.

Obtaining parking spot maps. One issue that may seem to limit large deployments is the effort to obtain parking spot maps, particularly when complicated time-dependent parking rules are in place. Beyond manual construction from satellite imagery, as in our project, maps may be available to some extend at city authorities. We believe, however, that maps can also be automatically generated through aggregation of sensor data over time periods of weeks to months. The intuition behind this idea is simple: spaces that almost *never* have cars parked are likely to be invalid parking spaces (driveways, storefronts, illegal parking spots such as fire-hydrants, etc., or simply portions where parking is not allowed), while spaces that *always* have a car parked are very likely not parked cars, but some other immovable object.

8. RELATED WORK

A number of approaches have been tested for the monitoring of parking spaces in recent times. Parking garages have been using in/out counters at the entry and exit points to count the number of additional vehicles they can accommodate at any given time [18]. This information is often displayed on prominent electronic signs-boards near such garages or on nearby roadways, allowing drivers to decide which way to go to find parking. Airports and rail stations have been using similar parking management systems in recent times [18]. A somewhat newer approach to finding and reserving parking spaces in urban areas, albeit with very limited success, has been tested by web-based markets such as those in [19, 20] that allow users to buy and sell parking spaces on the internet for specified times. Such an internet marketplace allows for owners of private spots (such as residential spaces) and parking garages to offer reservations to users looking for parking, and for people occupying public metered spaces to broadcast the time at which they will be vacating the parking spot. Needless to say, no reservation mechanism is possible for metered public spaces. Other web-based systems such as [21,22] allows travellers to access information about the availability of spaces at airports or rail stations and make prior reservations. In the academic literature, [23] demonstrated a toy system that monitors individual parking spaces using web cameras, allows users to query the system for vacant spaces on a web front-end and provides the user with the closest vacant parking space that meets the constraints entered by the user. Metered parking in the city of Chicago recently went through major changes [24] – individual parking meters were replaced with pay stations that handle a large number of parking spaces. Doing away with individual parking meters makes the detection of offending vehicles parked in unpaid spots by parking enforcement authorities harder. Using the slot-matching approach described in this paper, such detection can be greatly simplified. SFPark [8] is the only system in our knowledge that specifically monitors street-side parking spaces, albeit, at fairly high cost of operation and only for metered spaces and garages. In contrast, the mobile sensing approach proposed in this paper provides a probabilistic alternative at a potentially much lower cost, and can work for both metered slotted spaces as well as unslotted spaces.

Another line of work related to ours is the estimation of traffic conditions on roadways using mobile sensing. The authors in [2] propose techniques to mine location trace data reported by vehicles on streets reporting their locations infrequently over a cellular uplink, to characterize traffic patterns on road segments. VTrack [25] explores the use of commodity smartphones for localization using cellular and WiFi signals in addition to GPS for localization to estimate traffic delays on roadways with fine spatio-temporal granularity. Their work focuses on the challenges associated with energy consumption and the unreliability of sensors. Nericell [26] also utilizes sensors available on smartphones to detect road and traffic conditions in a city, using microphones to detect the level of honking and accelerometers to detect bumps and braking. Other work in mobile sensing includes [9] which uses the mobility of participating vehicles to detect potholes in roads using accelerometers, and aggregate this data over time to obtain the locations of the roads that are most in need of repair. [3] uses the concept of virtual triplines to address the privacy problems in traffic monitoring systems based on the reporting of GPS coordinates by individual users.

Prior work on getting location information in mobile systems has focussed heavily on the use of GPS receivers. Recent research on localization has also explored the use of GSM cellular signals for triangulation of cellphones [27]. Although cellular signals provide much lower location precision, they have also been used for traffic monitoring applications [25, 28, 29]. Finally, our work on improving location accuracy using the sensor in addition to the GPS, may be thought of as a locationing system with the fusion of information from multiple sensors, on which there exists extensive past literature (see for e.g. the location stack [30, 31] built by Hightower and Fox).

9. CONCLUSIONS

We have presented the ParkNet system, a mobile approach to collecting road-side parking availability information, which introduced more challenging location accuracy and sampling frequency requirements than earlier vehicular sensing applications. Based on over 500 miles of data collected over 2 months with our prototype vehicles, we showed that ultrasound sensors combined with GPS can achieve about 95% accurate parking space counts and can generate over 90% accurate parking occupancy maps. To create occupancy maps, we corrected GPS errors using environmental references points, since we found that GPS error were correlated over time and space even with WAAS support. Using trace-driven simulations from San Francisco taxicabs we showed that equipping only 536 vehicles with sensors would lead to a mean inter-sampling interval of about 25min in 85% of the downtown area, or about 10 min in 80% of the area. Further, we expect that the density of taxicabs in an urban area to be strongly correlated with the presence of street parking spaces, since both are driven by the presence of a large number of people. With the small number of sensors required, the mobile sensing approach therefore promises significant cost benefits over current stationary sensing approaches—

by an estimated factor of 10-15. We expect that the environmental positioning approach and the taxicab coverage analysis will also benefit other mobile sensing applications.

10. ACKNOWLEDGEMENTS

We would like to thank Ivan Seskar for his constant guidance and support of our efforts, and Ilya Chigirev for his help in building our magnet-mounted sensor prototypes. We would also like to thank Mesut Ali Ergin for helping out with vehicular installation and wiring of mobile ORBIT nodes. The material presented in this paper is partially based upon work supported by the National Science Foundation under Grant No. CNS 0627032.

11. REFERENCES

[1] "Urban Mobility Report, *Texas Transportation Institute*, Texas A&M University," 2007.

[2] J. Yoon, B. Noble, and M. Liu, "Surface street traffic estimation," in *MobiSys '07: Proceedings of the 5th international conference on Mobile systems, applications and services*. ACM, 2007, pp. 220–232.

[3] B. Hoh, M. Gruteser, R. Herring, J. Ban, D. Work, J.-C. Herrera, A. M. Bayen, M. Annavaram, and Q. Jacobson, "Virtual trip lines for distributed privacy-preserving traffic monitoring," in *MobiSys '08: Proceeding of the 6th international conference on Mobile systems, applications, and services*. ACM, 2008, pp. 15–28.

[4] D. Shoup, "Cruising for parking," *Access*, vol. 30, pp. 16–22, 2007.

[5] ——, "The Price of Parking on Great Streets," in *Planetizen Contemporary Debates in Urban Planning, Washington, DC*, Island Press, pp. 52-56., 2007.

[6] "Streetline Networks, http://www.streetlinenetworks.com."

[7] "http://www.vehiclesense.com."

[8] "http://sfpark.org/."

[9] J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, and H. Balakrishnan, "The pothole patrol: using a mobile sensor network for road surface monitoring," in *MobiSys '08: Proceeding of the 6th international conference on Mobile systems, applications, and services*. ACM, 2008, pp. 29–39.

[10] "Advanced parking management systems: A cross-cutting study – taking the stress out of parking, federal highway administration, intelligent transportation systems, u.s. department of transportation, jan 2007."

[11] H. L. V. Trees, *Detection, Estimation, and Modulation Theory: Part I*. Wiley-Interscience, 2001.

[12] E. Kaplan and C. Hegarty, "Understanding GPS, Principles and Apllications," in *Artech House*, ISBN 1-58053-894-0.

[13] "Garmin GPS 18, Technical Specifications, www.garmin.com/manuals/425_technicalspecification.pdf."

[14] R. Burkard, M. Dell'Amico, and S. Martello, "Assignment Problems," in *SIAM*, ISBN 978-0-898716-63-4, 2009.

[15] H. W. Kuhn, "The Hungarian Method for the assignment problem," in *Naval Research Logistics Quaterly 2:83-97*, 1955.

[16] "Yellow cab of san francisco, location dataset, http://cabspotting.org/," 2008.

[17] "Press release - sfmta completes innovative citywide census of san francisco parking, san francisco municipal transporation agency, http://sfpark.org/media/."

[18] "Smart-parking at rockridge BART station, http://www.path.berkeley.edu/path/research/featured/120804/smart-park.html."

[19] "http://www.spotscout.com."

[20] "https://www.mobileparking.com."

[21] "http://www.aboutairportparking.com/."

[22] "http://www.parkingcarma.com."

[23] J. Campbell, P. B. Gibbons, S. Nath, P. Pillai, S. Seshan, and R. Sukthankar, "Irisnet: an internet-scale architecture for multimedia sensors," in *MULTIMEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia*. ACM, 2005, pp. 81–88.

[24] "Chicago Parking Meters, http://www.chicagometers.com."

[25] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson, "VTrack: accurate, energy-aware road traffic delay estimation using mobile phones," in *SenSys '09: Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*. ACM, 2009, pp. 85–98.

[26] P. Mohan, V. N. Padmanabhan, and R. Ramjee, "Nericell: rich monitoring of road and traffic conditions using mobile smartphones," in *SenSys '08: Proceedings of the 6th ACM conference on Embedded network sensor systems*. ACM, 2008, pp. 323–336.

[27] A. Varshavsky, "Are GSM phones THE solution for localization?" in *WMCSA '06: Proceedings of the Seventh IEEE Workshop on Mobile Computing Systems & Applications*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 34–42.

[28] "http://www.airsage.com."

[29] "http://www.intellione.com."

[30] J. Hightower, D. Fox, and G. Borriello, "The location stack," University of Washington, Department of Computer Science and Engineering, Seattle, WA, UW CSE 03-07-01, July 2003.

[31] J. Hightower and G. Borriello, "The location stack: Multi-sensor fusion in action," Demonstration at the Fifth International Conference on Ubiquitous Computing (Ubicomp), October 2003.