ParkNet: Drive by Sensing of Parking Spots
Background

- Automotive traffic congestion imposes significant societal costs

- A $78 billion annual drain on the U.S. economy in 2007 in the form of 4.2 billion lost hours and 2.9 billion gallons of wasted gasoline [“Urban Mobility Report, Texas Transportation Institute, Texas A&M University,” 2007]

- Vehicles looking for parking wastes about 95,000 hours (eleven years) of drivers’ time every year and cruises 950,000 miles (38 trips around the world), burning 47,000 gallons of gasoline and producing 730 tons of carbon dioxide. [D. Shoup, “Cruising for parking,” 2007]
Key Factors

• Lack of information about roadside parking availability

• Road-side parking spots are commonly underpriced compared to parking garages
Key Contributions

• Demonstrating the feasibility of a mobile sensing approach to road-side parking availability detection through the design, implementation, and evaluation of such a system

• Proposing and evaluating an improved approach to GPS positioning using environmental fingerprinting the location accuracies necessary for precise matching of cars with their associated parking slots

• Showing through trace-based simulations with a dataset that a few hundred taxis provide adequate spatiotemporal sampling of a down-town area
Parking Information - 1

- **Slotted areas**: Areas where vehicles are arranged in slots often separated by lines marked in the road.

- **Unslotted areas**: Areas without any marked space.
  - $d_i$: The distance of all available spaces of valid parking
  - $d_{\text{spot}}$: A fixed size of parking spot (typically 6m)
  - $n = \frac{d_i}{d_{\text{spot}}}$
Parking Information -2

• Two types of parking information, “Space Count” and “Occupancy Map”
• Space Count: No of parking spaces available on one given road segment
• Occupancy Map: A map showing each parking slot as occupied or not
System Design Goals and Requirements

• **Provide Parking Statistics**
  - The system should be able to determine the availability of road-side parking spaces

• **Assist Parking Enforcement**
  - The system should be able to identify parking spots occupied by an infringing vehicle

• **Low-cost Sensors**
  - The system should operate with sensors that are typically used in automobiles for other applications. (ultrasonic sensors)

• **Low vehicle participation rates**
  - The system should be able to operate with even low vehicle participation rates
System Deployment - 1

- Three vehicles which collected parking data over a 2 month time frame during their daily commute

- 3 road-side parking areas in Highland Park, New Jersey
  - One of them is 57 marked slotted parking spots
  - Two of them are 734 m and 616 m areas in length

- More than 500 miles of data on streets with parking were collected

- The data collection was not controlled in any manner (e.g. speed, traffic conditions, obstacles, etc.)
System Deployment - 2

- 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.

- A Garmin 18-5Hz GPS with 12 channel receiver (provides 5 fresh GPS readings per second, and a real-time WAAS correction of errors less than 3 meters) was used.

- Both the sensor and the GPS provide data in serial format, which can be accessed via an USB serial port on a computer.
The mounted sensor (Maxbotix WR1) 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 (12 inch -> the nearest obstacle, 255 -> no obstacle).
A Sony PS3 Eye webcam was integrated into the sensor mount for system evaluation purposes.
System Deployment - 4

Parking estimation server

<Kernel-time, range, latitude, longitude, speed>
Tripbox: Limiting Data Collection -1

- Data collection is limited to parking regions due to small areas of roadside on one commute trip and large volume of video data involved
- The activation and deactivation of data collection is implemented using “Tripbox”
- The tripbox represents rectangular areas defined by latitude, longitude
Tripbox: Limiting Data Collection -2

- An entry function -> starts data collection
- An exit function -> stops data collection

- The tripbox daemon 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
  - If the mobile node is already inside the tripbox and the next received coordinate is outside this region, it triggers the exit function
Tripbox: Limiting Data Collection - 3

- The tripbox implements *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.
Detection Algorithms - 1

One dimensional view of the distance to the nearest obstacle
Detection Algorithms - 2

- **Slotted model**
  - All dips that have too few readings and could not possibly have arisen from a parked car (<6 sensor readings, assuming a max speed of 37mph and car length of 5m)
  - To detect a parked car, the width and depth of each dip in the sensor reading is compared against thresholds (89.7 inches for the depth and 2.52 meters for the width with)
  - Finally, all remaining dips are checked for spatial width, and compared against a threshold representing the typical length of a car
Detection Algorithms - 3

• **Unslotted 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.
  
  - The spatial distance between dips that have been classified as parked cars are estimated.
  
  - The estimated length of the vacant stretch is then compared against the length of a standard parking space (6m)
Challenges in Ultrasonic Sensor and GPS

• An ultrasonic sensor does not have a perfectly narrow beam-width

• The sensor receives echos not just from objects that are directly in front but also from objects that are at an angle

• A commercial grade GPS receiver suffers errors (inaccuracy of latitude and longitude values)

• As a consequence, GPS errors can make a parked car appear to be shorter or longer than its true length
Error Rate of Slotted Model

• 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.

• \( n \) = Estimated number of slots by a sensing vehicle

• \( n \) = The number of vacant slots

• The ratio, \( \frac{n}{n} \) = The performance of the detection

• The ratio can be smaller or greater than 1 for a given run

• The ratio is expected to have a mean close to 1
Error Rate of Unslotted Model

- The algorithm records an estimated space, whenever a space between two parked cars is ascertained to be large enough to accommodate another car.

- \( d \) = Recorded estimated space

- \( d \) = The actual space between cars

- The ratio, \( d/d \) = The performance of the detection
ParkNet

PART II

Presenter’s Name: Jiangliang Cai
Evaluation of detection algorithms

- Slotted mode: threshold of 2.5 meters
- Unslotted mode: estimated space compared with 6 meters
- 96% accurate in terms of obtaining parking counts
Occupancy map creation

GPS error correlation

- 29 different runs, 8 objects
- Deviation error:
  - in 4.6 m on X-direction
  - in 5.2 m on Y-direction
- Street is narrow to allow variation within ±1/2 meter
Occupancy map creation

Environmental fingerprinting

- Some fixed objects in the environment are well known
- Using the first object to correct the error of the remaining 7 objects
- Location error builds up with increasing distance from the object 1
Occupancy map creation

**Slot-matching**
- 57 slots on a street were determined by Google Earth
- Using slotted model
- Using MATLAB

![Bar chart showing percentage errors](chart.png)
Case study of San Francisco

Description of case study

- Viability of ParkNet related to number of mobile sensors
  - Should be enough to “monitor” street-parking space
  - Important to determine how often a node would pass by on a randomly street

- Should find a balance to control cost

- 536 taxicabs in San Francisco collected over a roughly one month during 2008
Case study of San Francisco

- Consider two geographical areas
- Cabs spend most time in smaller area
Case study of San Francisco

Cost analysis

(a) Empirical CDF of mean time between cabs (min)

(b) Empirical CDF of mean time between cabs (min)
Case study of San Francisco

Case evaluation

- The operational running cost comprises maintenance and communications costs.

- Total cost of ParkNet less than a fixed sensor system

- Whereas fixed sensor provide continuous monitoring

- The cost benefits of ParkNet are still significant
Related work

Other approaches

- Parking garages using in/out counters at the entry and exit points
- Web-based market
- City of Chicago using pay station
- One approach related to ours is using mobile sensing
  - reported by vehicles on streets
  - Using commodity smartphones
Future work

Weakness and difficulty to improve

- Power source: convert the 12 volt DC vehicle power supply to AC power

- Multilane roads

- Speed limitations: the sensor provide only 20 distance readings per second

- Obtaining parking spot maps: spaces that almost never have cars parked or always have a car parked
Future work

Future of ParkNet

- Higher rate of change in process monitored
- Higher accuracy
- Fuel and time will be more and more precious
- Market is in big or mega city which have serious congestion problem
- Combine with web or smartphone
Conclusions

Problem
- Lack of information about drive-by parking
- Drive-by parking monitoring

Critical assessment of contribution
- ParkNet achieve to detect drive-by parking with high accuracy
- Environmental fingerprinting improve accuracy of GPS positioning
- With quite low number taxis, adequate to achieve precisely where parking is badly need

Weakness
- The biggest weakness of ParkNet is the speed limit

Will it be popularized?