Crowd and Mobile Sensing

Using Mobile Phones as Sensors

CSE 40437/60437-Spring 2015

Prof. Dong Wang

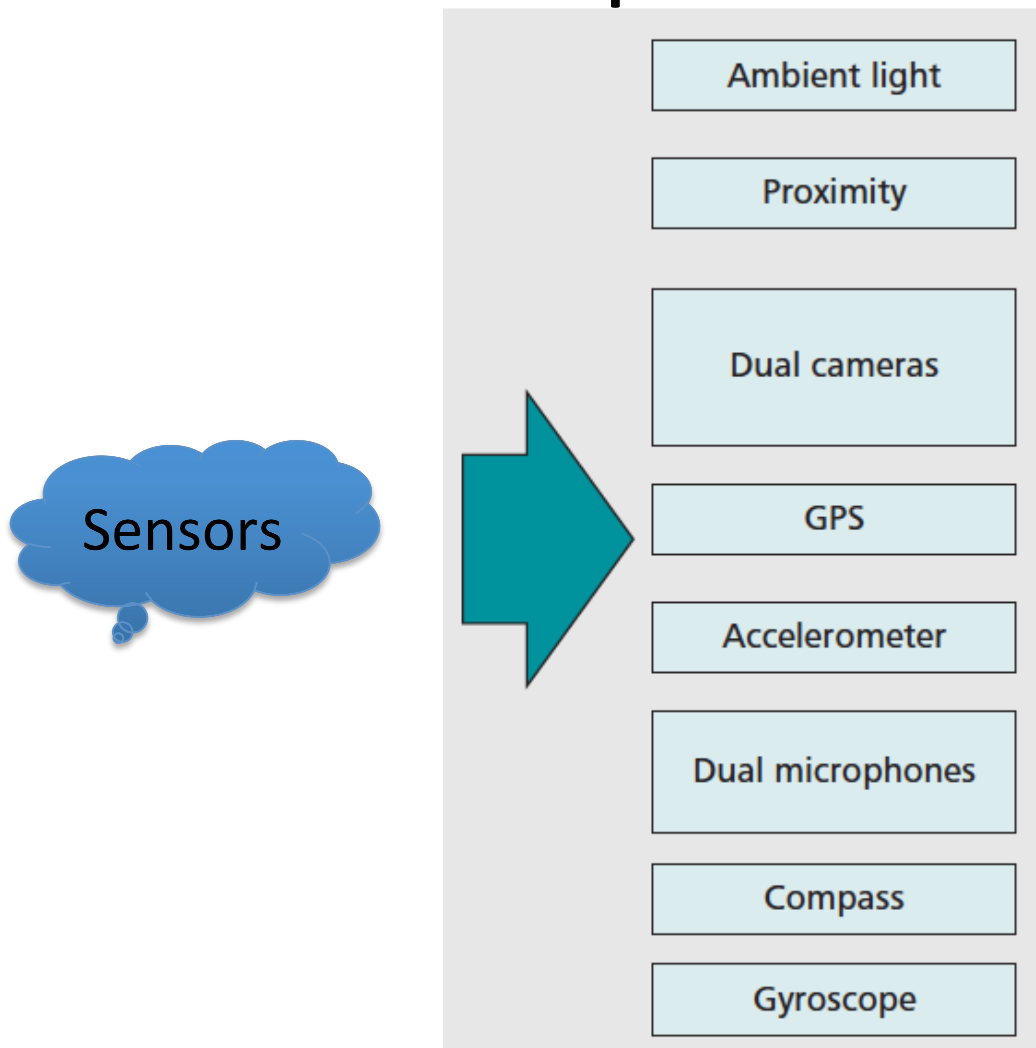
Papers

- Paper 1: "A survey of mobile phone sensing." Lane, Nicholas D., et al. Communications Magazine, IEEE 48.9 (2010): 140-150.

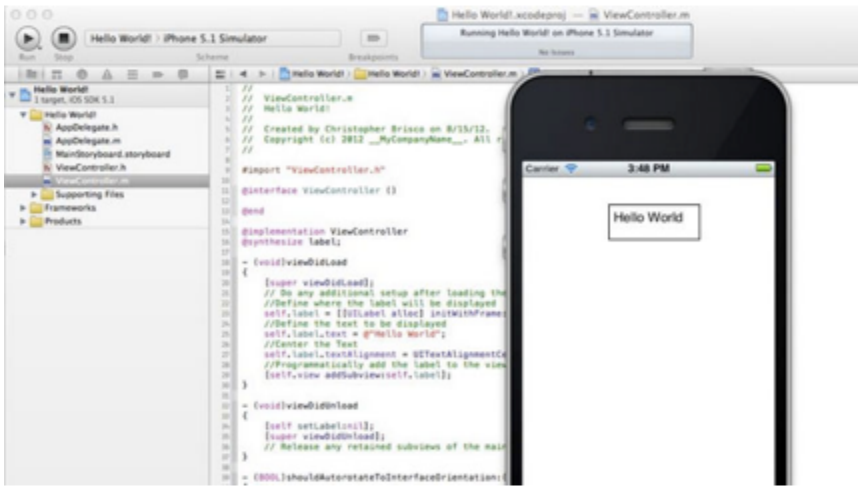
From Mobile (Smart)phones to CrowdSensing

What makes this happen?

Technical Enabler 1: Powerful Embedded Sensors in Smartphones



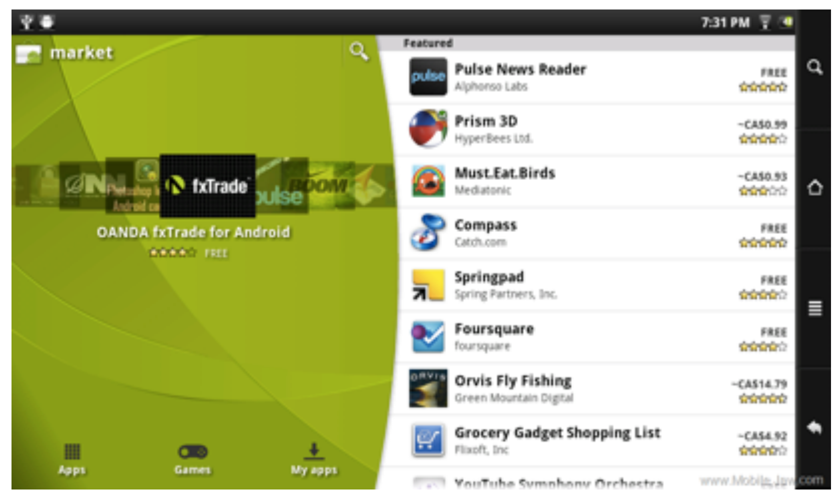
Technical Enabler 2: Smartphones are open and programmable



Technical Enabler 3: Phone vendors now offer ***app store*** to delivery new apps

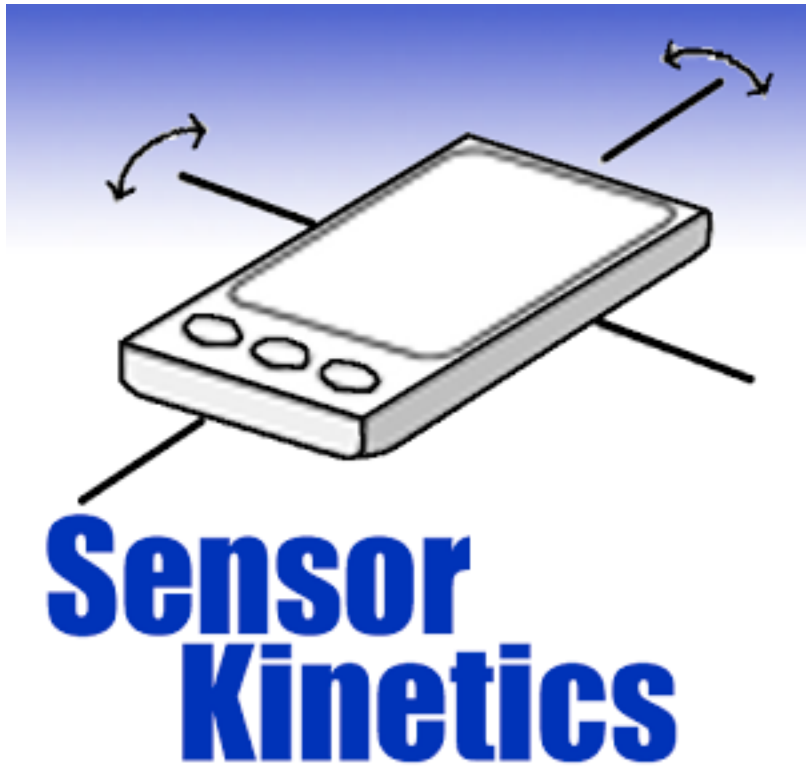
Apple Store

ANDROID APP ON Google play



Technical Enabler 4: Mobile Computing
Cloud-> offload services to back-end

Demo: Sensors on Android Phones



https://play.google.com/store/apps/details?id=com.innoventions.sensorkinetics&hl=en

Demo: Sensors on Android Phones

https://play.google.com/store/apps/details?id=imoblife.androidsensorbox&hl=en

What can phone sensors do?

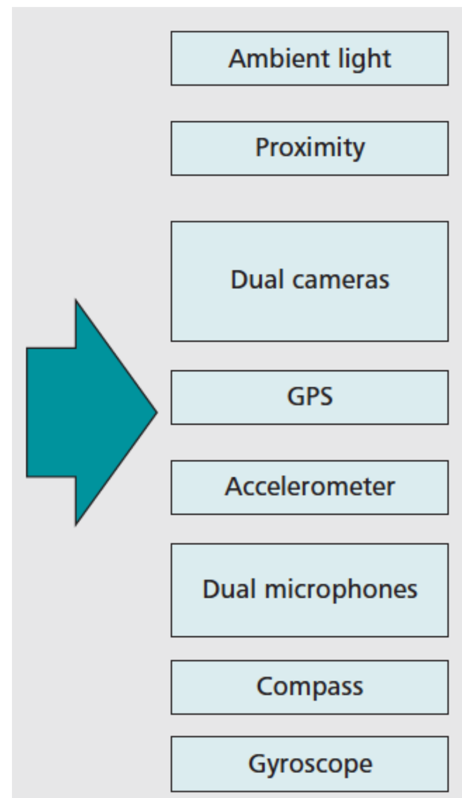
- Proximity sensors:
  - Detect when the user holds her phone close to face -> disable touchscreen
- Lightness sensors:
  - Adjust the brightness of screen to save power
- GPS: identify phone location:
  - Local search, mobile social network, navigation
- Compass and gyroscope:
  - provide direction and orientation in location-based apps

What can phone sensors do?

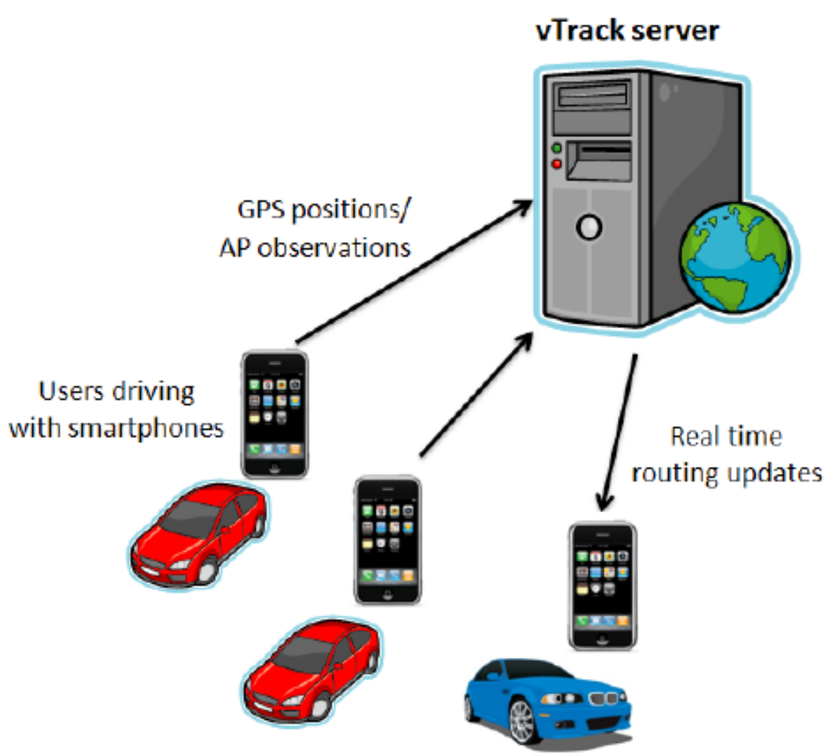
- Accelerometers:
  - Characterize physical movements of users; Activity recognition (e.g., running, walking, standing).
- Camera and Microphone:
  - Record personal digital trace. Context Sensing (e.g., where is the user and what she is doing now)
- Combination of accelerometer and GPS/Celluar signal:
  - Recognize the mode of transportation (e.g., bus vs subway)
- WiFi and Bluetooth:
  - Indoor localization and detecting social contact

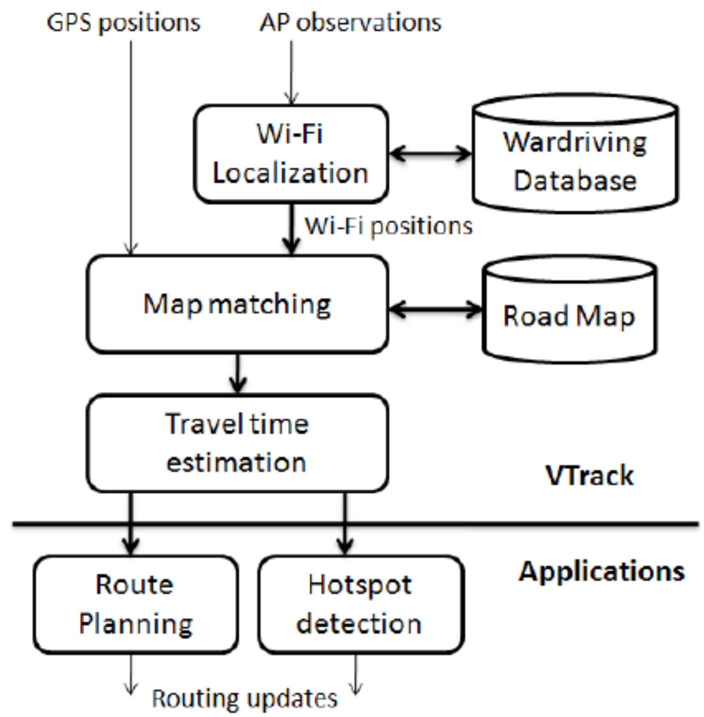
Applications

- What are the interesting applications you can think of using one or a set of sensors available on the smartphones?



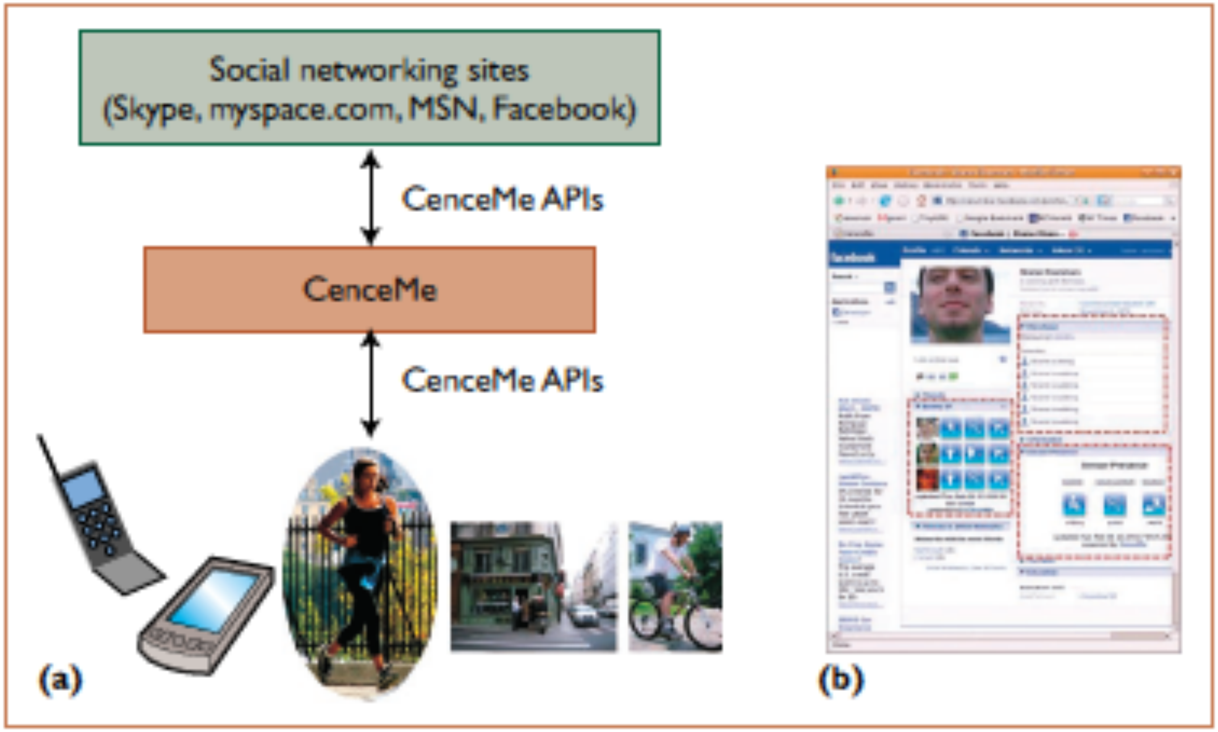
Applications: Transportation





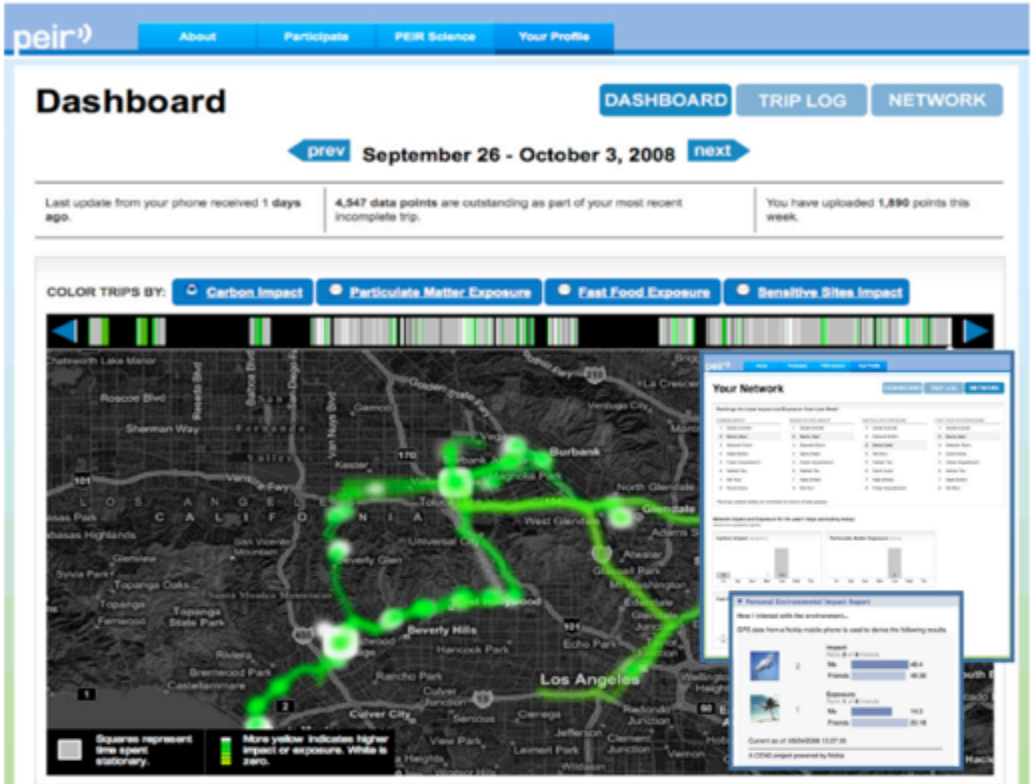
MIT VTrack: Use GPS and WiFi signals on driver's smartphones to estimate delay prone segments on city streets.

Applications: Social Networking



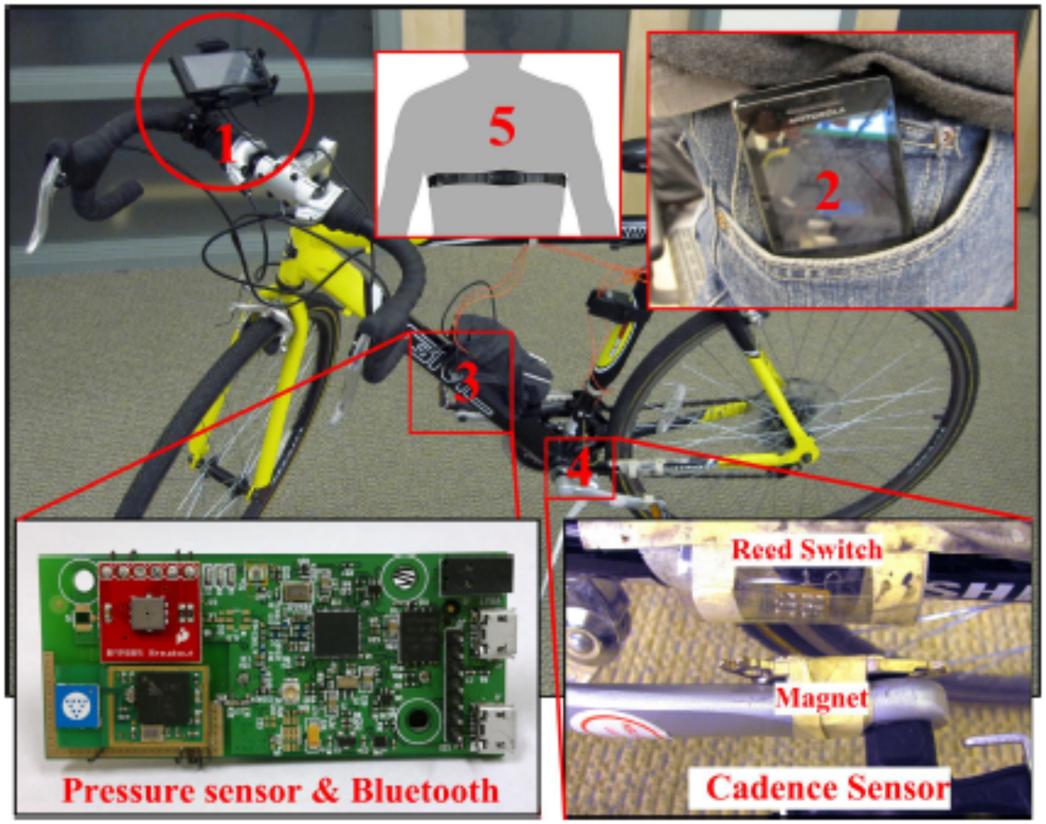
Dartmouth CenceMe: Use sensors on smartphones to automatically classify events in people's lives ("where are u and what are u doing?") and selectively share it on online social networks (e.g, Twitter, Facebook, etc.)

Applications: Environment Monitoring



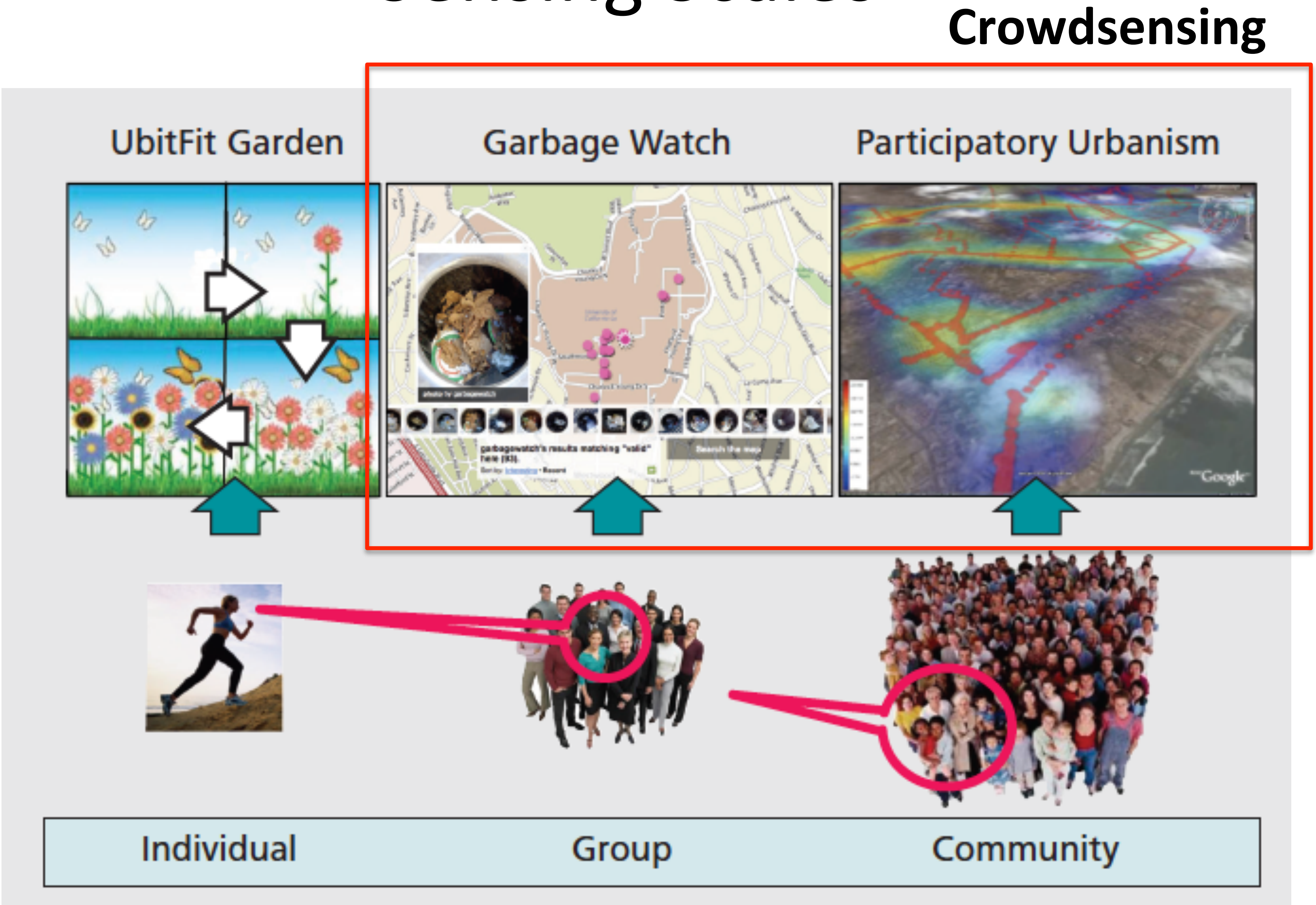
UCLA Peir: A personal environment impact report that uses sensors on phones to track how the actions of individuals affect their exposure and contribution to environmental problems (e.g., carbon emissions)

Applications: Health and Wellbeing



Johns Hopkins Pocket Sensing: Use sensors on the smartphone in the bicyclist's pocket to accurately estimate measure her cadence and caloric expenditure.

Sensing Scales



Sensing Paradigms

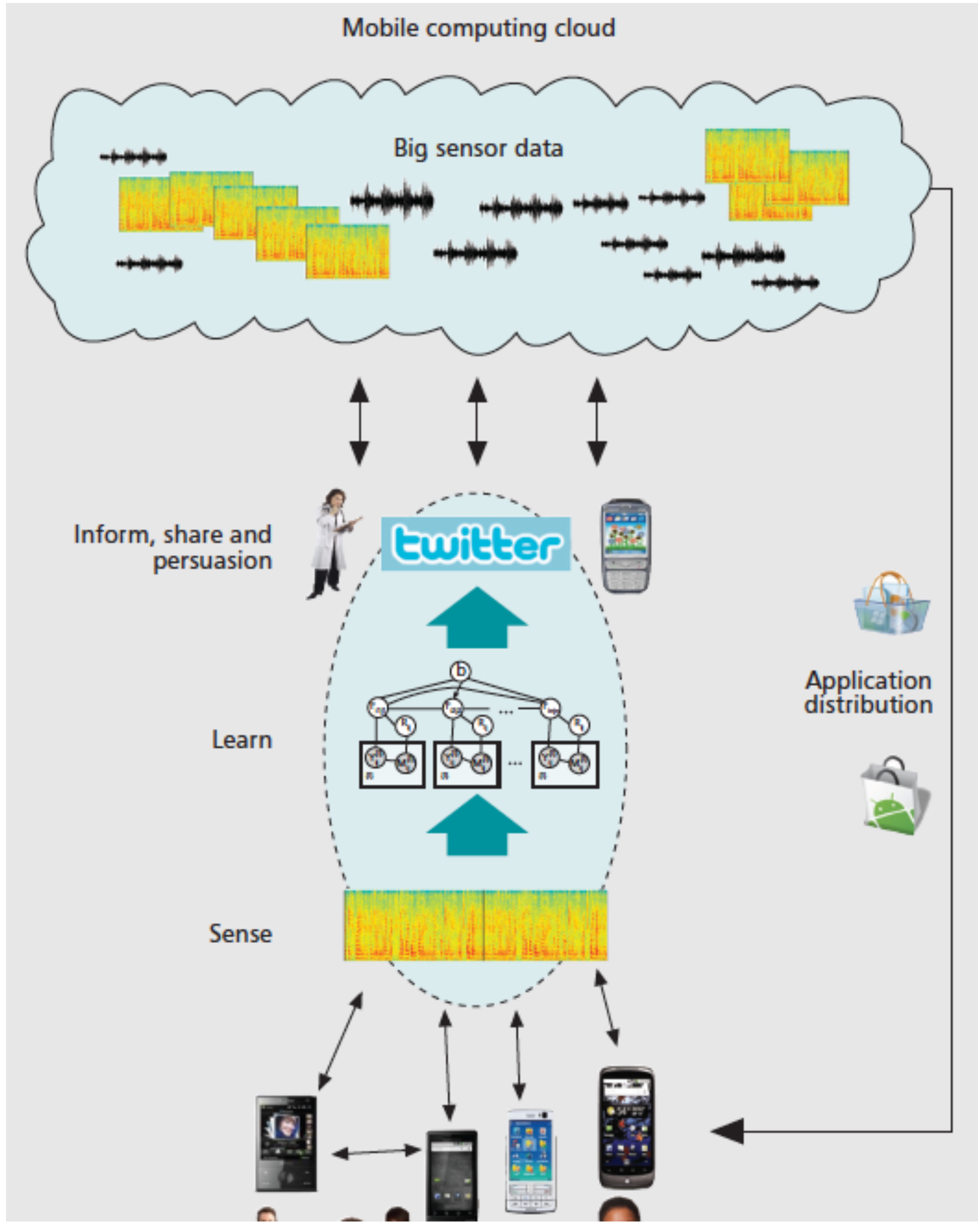
- Participatory Sensing
  - Users actively engage in the “sensing process”
  - Human intelligence can be leveraged for complex tasks
  - More costs or incentives are needed to keep humans involved
  - Privacy Issues
- Opportunistic Sensing
  - Fully automated and no user involvement
  - Less burden and costs on the user
  - Detect the phone context
  - Humans are underutilized
  - Privacy and Energy Issues

Sensing Architecture

Close the Loop

Learn

Sense

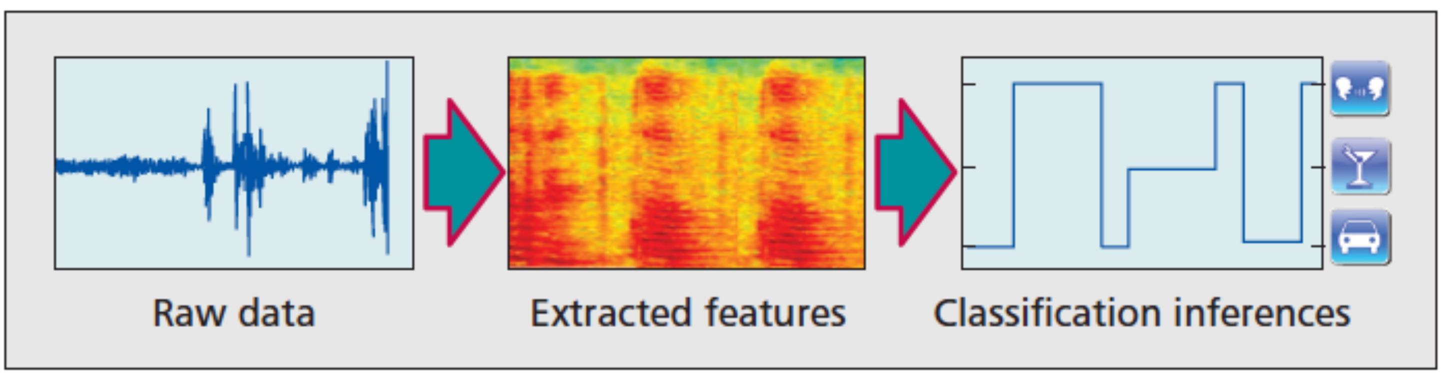


Sense

- Programmability:
  - Lack of low level sensor control
  - Different vendors offer different APIs
- Continuous sensing:
  - Need to support multitasking and background processing
  - Limited battery power on mobile phones
- Phone Context:
  - Phones are used on the go and in different contexts (e.g., in vs out of pocket)
  - Anticipating all possible different phone usage scenarios is very difficult

Learn

- Human Behavior and Context Modeling
  - Supervised learning (small scale)
  - Semi-supervised/Unsupervised (medium to large scale)
  - Learn every data activities (e.g., brushing teeth, driving, running)
  - Learn places (e.g., work, home, coffee shop)



Close the loop

- Sharing
  - Standardized method: Visualization using a **web portal** (e.g., display sensor data and inferences)
  - Leverage **social media** outlets (e.g., Twitter, Facebook, Flickr) to build a community around a sensing application (e.g., Nike+)
- Personalized Sensing
  - Monitor individual's daily activities and profile their preferences (e.g., voice search on Google)
  - Make personalized recommendations (e.g., book, clothes, food, etc.)

Close the loop

- Persuasion
  - Peer pressure, sharing the sensed data or information within a community or social network
  - Design interesting interface that targets user's individual goal (e.g., UbiFit)
- Privacy
  - Key concern for people to participate and share their data (which can reveal a LOT of information)
  - Local data processing and aggregation
  - Adding controlled random noise that does not affect aggregated results (e.g., GreenGPS)

Open Questions

- How much intelligence we shall push to the phone without jeopardizing the phone experience?
- How do we scale the sensing application from individual to a large community?
- How to efficiently process and storage the big data from the mobile and crowdsensing apps?
- How to efficiently filter noises from the collected data, especially when humans are in the loop?

Papers

- Paper 2: How Long to Wait? Predicting Bus Arrival Time with Mobile Phone based Participatory Sensing. Zhou, Pengfei, Yuanqing Zheng, and Mo Li. Proceedings of the 10th international conference on Mobile systems, applications, and services (Mobysis 12). ACM, 2012.

Goal

- **Goal:** Predict bus arrival time accurately using collaborative efforts from crowds

- **How long to wait ?**
  - Alternative transit choices
  - Better travel plans

Q: What is current solutions to predict bus arrival time?

Exist Solutions

BUS TIMETABLE

We have our very own Fort Dunlop shuttle bus to run you to and from the city centre, so you don't have to drive. This is good for the environment and takes a few more cars off the road.

The bus is fitted with a few techno gadgets. It's got on board Wi-Fi access so you can now surf the web in style. It's also fitted with sat nav. It means you can keep track of its movements from behind your desk or at home if you love it that much.

Fort Dunlop		0805	0905	1005	1105	1205	1305	1405	1505	1605	1705	1805	1905
The Fort Shopping Park		0810	0910	1010	1110	1210	1310	1410	1510	1610	1710	1810	1910
Star City (Watson Rd)		0813	0913	1013	1113	1213	1313	1413	1513	1613	1713	1813	1913
Millenium Point (Jennens Road)		0820	0920	1020	1120	1220	1320	1420	1520	1620	1720	1820	1920
Corporation Street	0725	0825	0925	1025	1125	1225	1325	1425	1525	1625	1725	1825	1925
Navigation Street (for Mailbox)	0729	0829	0929	1029	1129	1229	1329	1429	1529	1629	1729	1829	1929
Birmingham New Street station (St Martin's Circus Queensway)	0735	0835	0935	1035	1135	1235	1335	1435	1535	1635	1735	1835	1935
Millenium Point (Jennens Road)	0740	0840	0940	1040	1140	1240	1340	1440	1540	1640	1740	1840	
Star City (Watson Rd)	0745	0845	0945	1045	1145	1245	1345	1445	1545	1645	1745	1845	
The Fort Shopping Park	0750	0850	0950	1050	1150	1250	1350	1450	1550	1650	1750	1850	
Fort Dunlop	0755	0855	0955	1055	1155	1255	1355	1455	1555	1655	1755	1855	

FORT DUNLOP

– Timetable (operating hours, time intervals, etc.)

Cons: Static and Not timely updated

Current Solutions

– Complex information system with special in-vehicle GPS devices

Cons: Substantial costs; Collaborative bus operators; Local availability; "1 min" != 1 min

Design Goals

- Crowdsensing approach
- Independent of transit operators
- No in-vehicle GPS devices (GPS signals are not always good in big cities)
- Energy Efficient
- Fully automatic

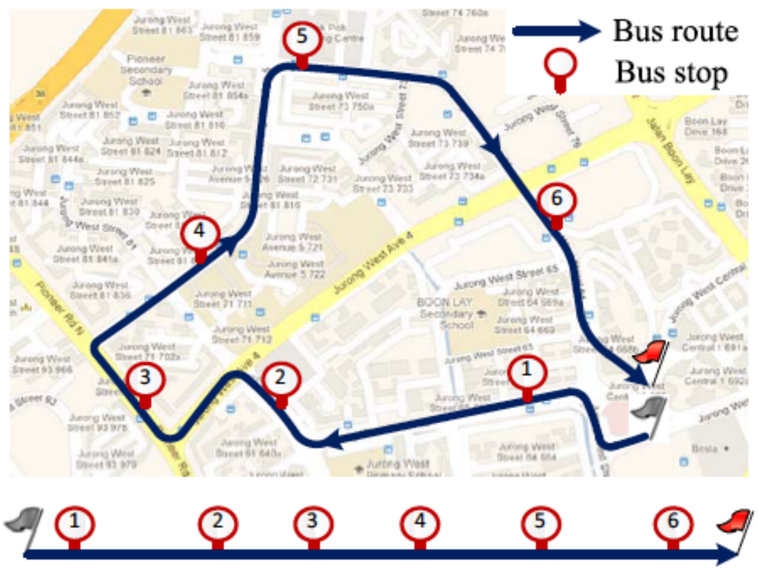
Share your thoughts …

- How would you solve the problem by designing a crowd-sensing application?

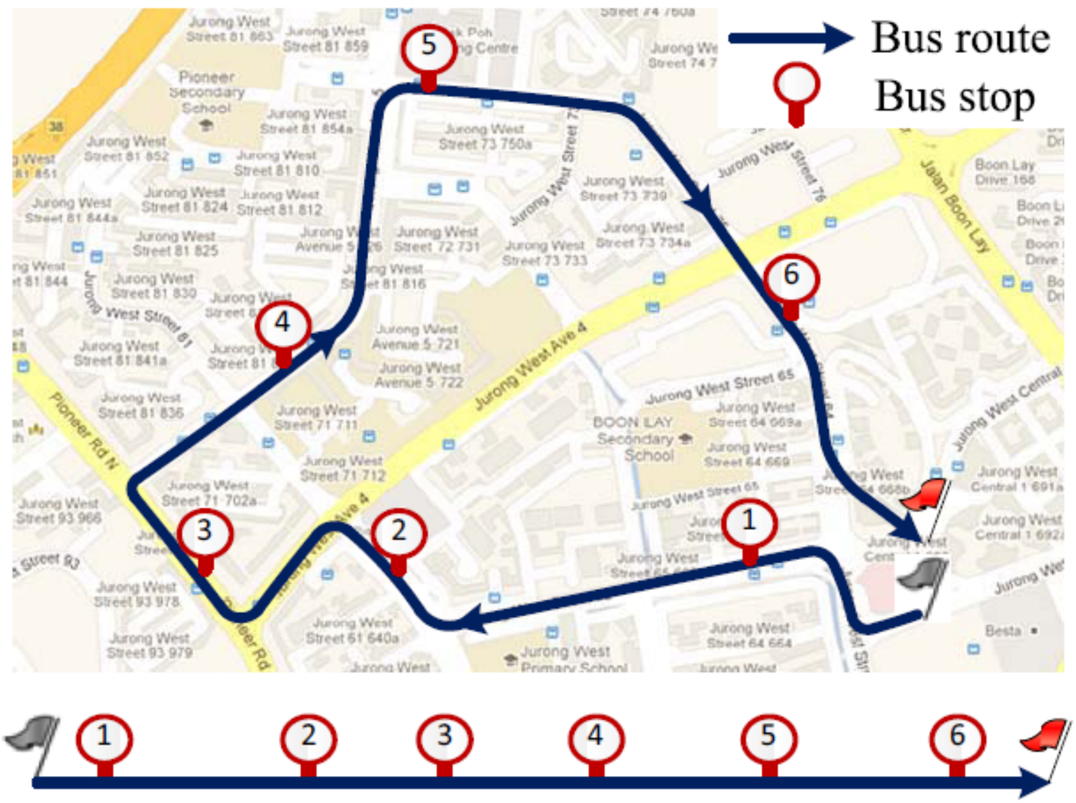
- What are the design challenges you have in mind?

Their Solution

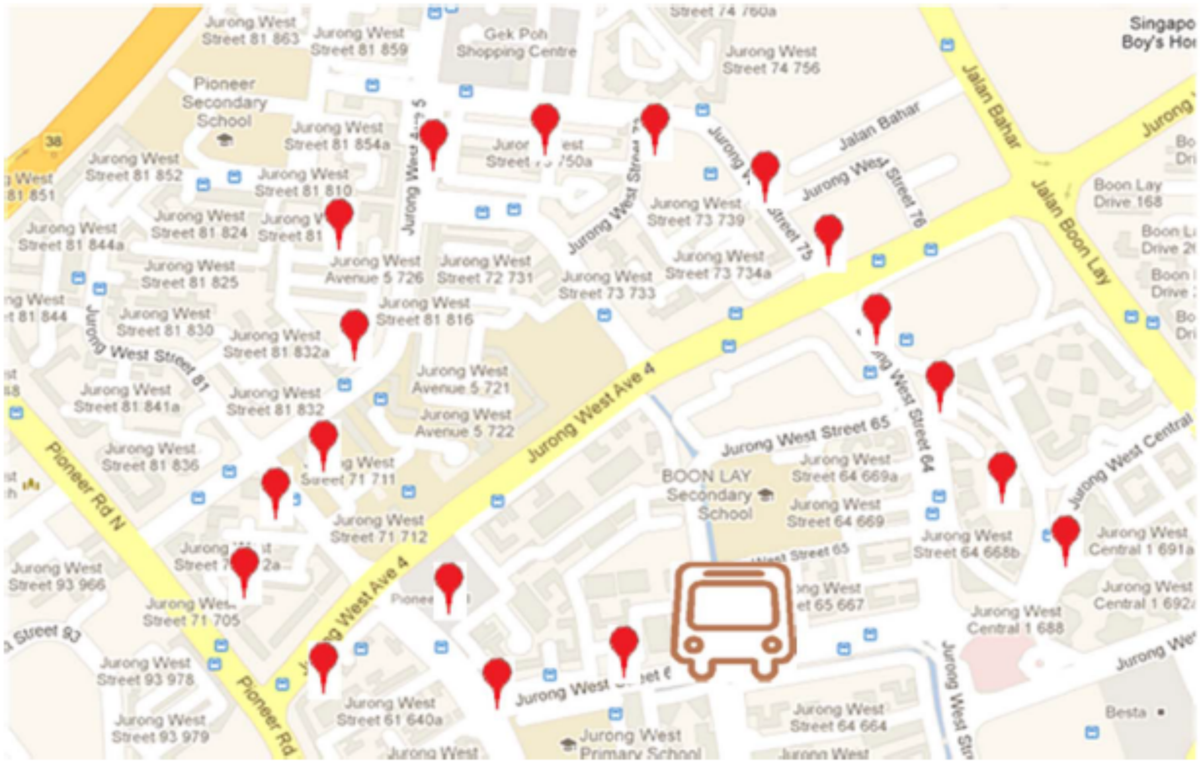
- Use the **cellular signals** of passenger's **mobile phones** to predict bus arrival time.



Key Question: How could we track bus location in real-time in a 2D space?

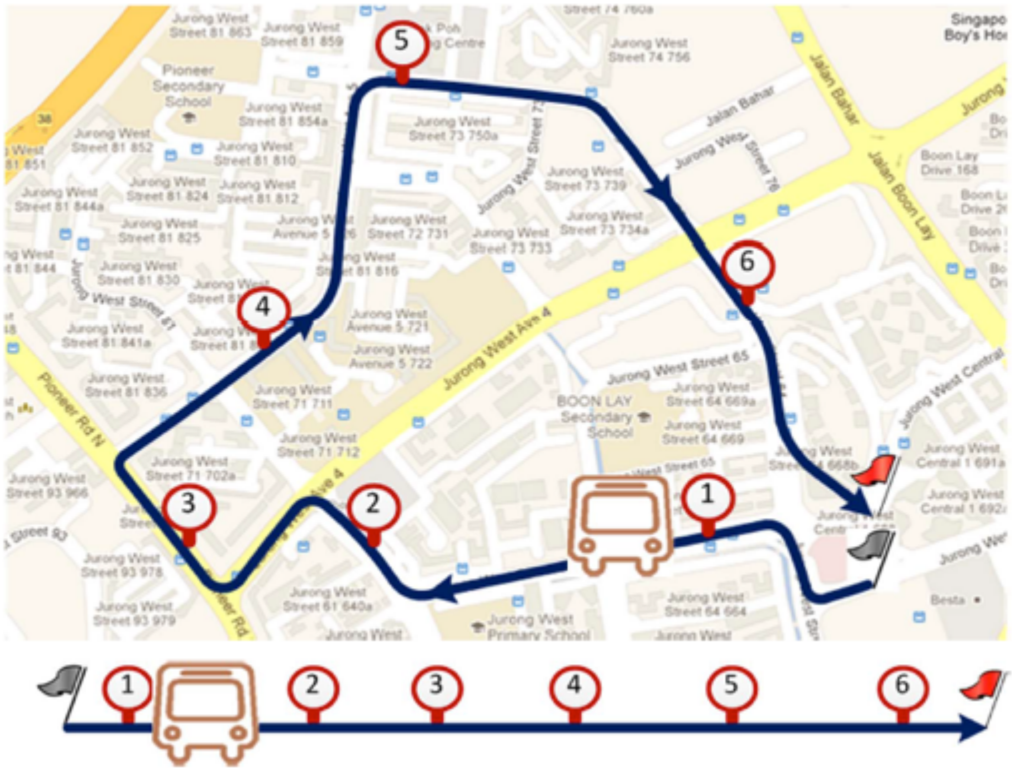


Basic Q: 2D vs 1D



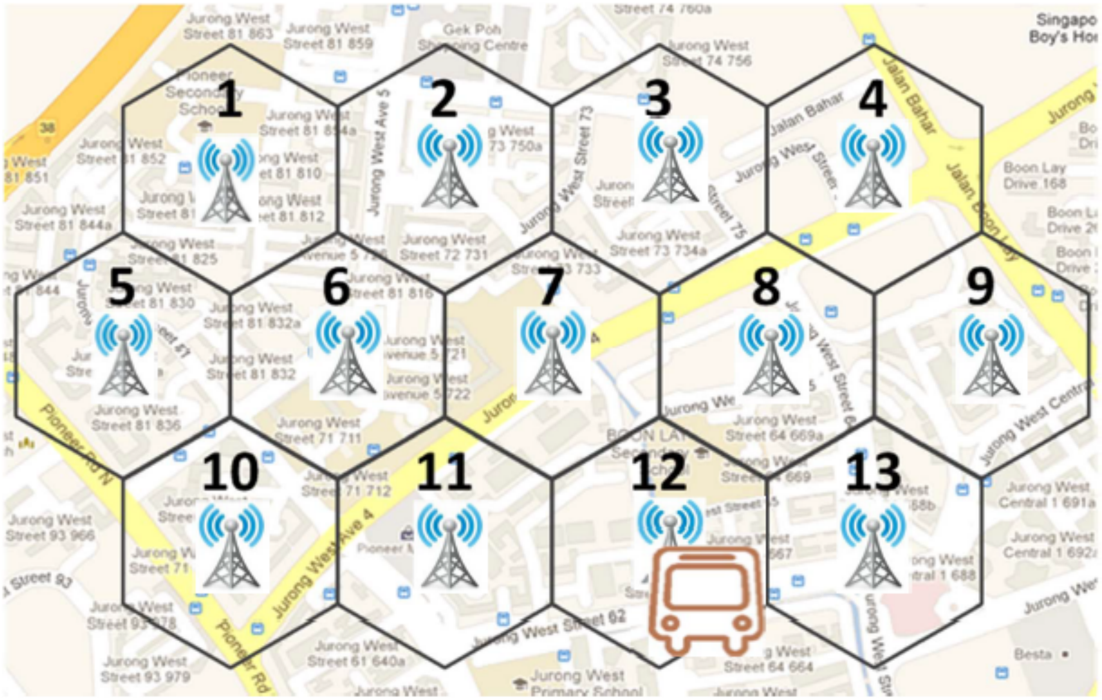
- City map is a 2D (dimension) space

Basic Q: 2D vs 1D



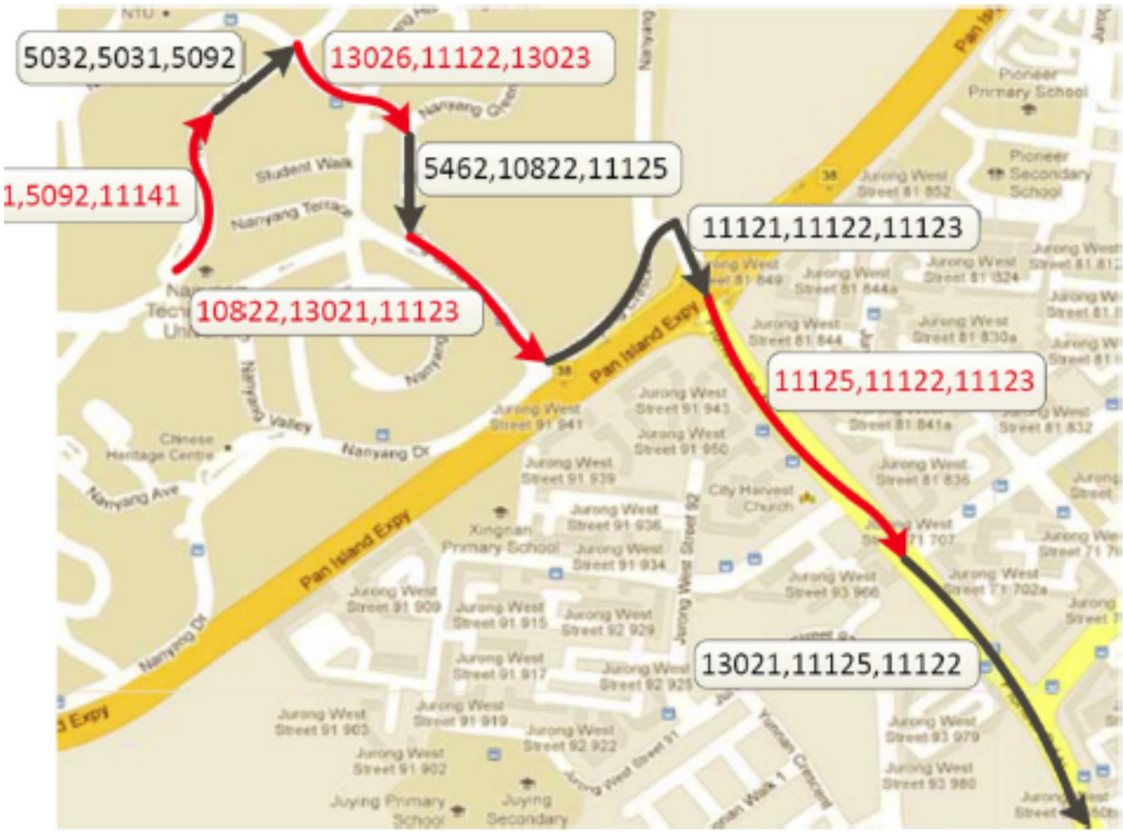
- Bus route is simply 1 D space

In cellular space



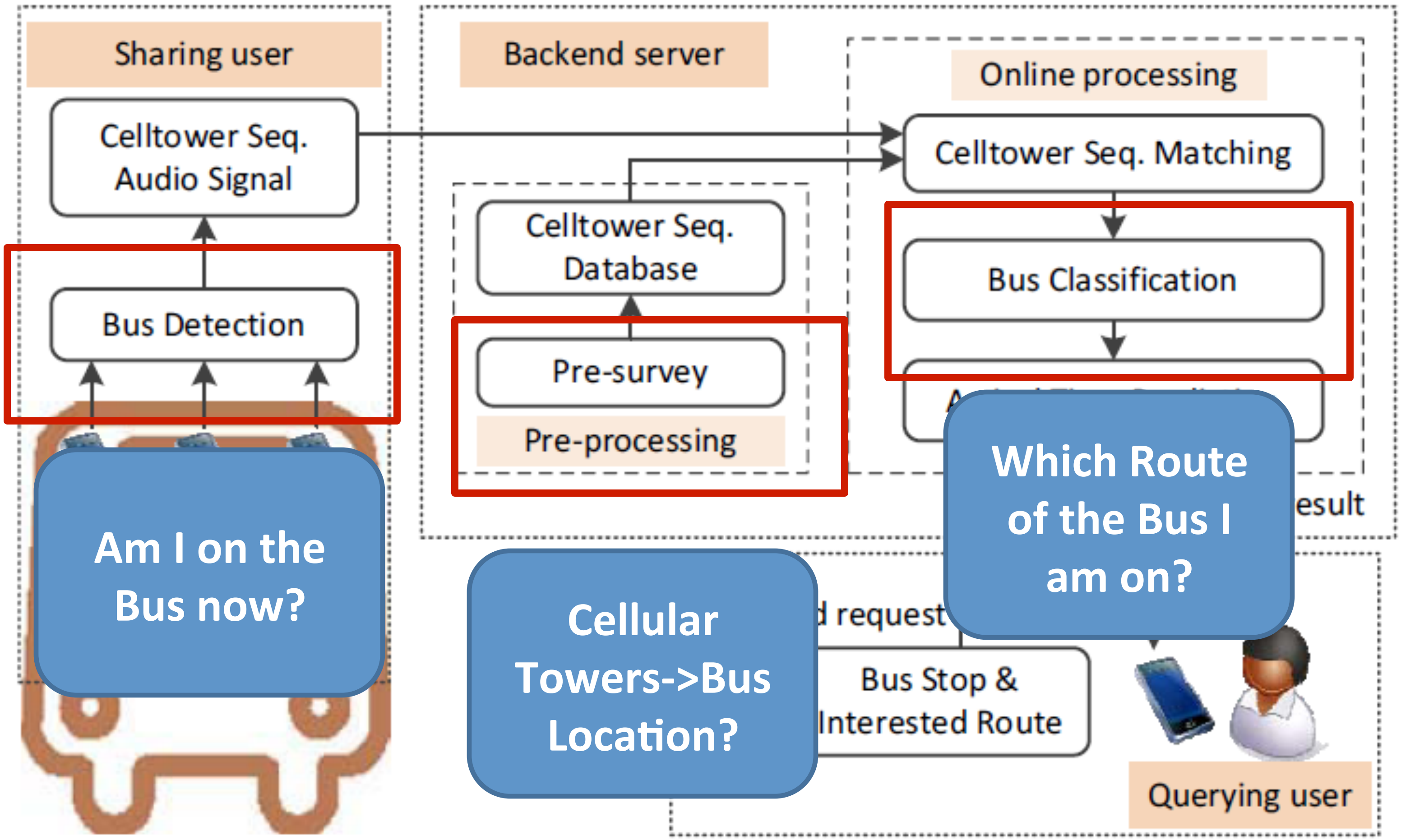
- The bus route can be characterized by a sequence of cells the bus goes by

Mapping bus route to cell tower ID



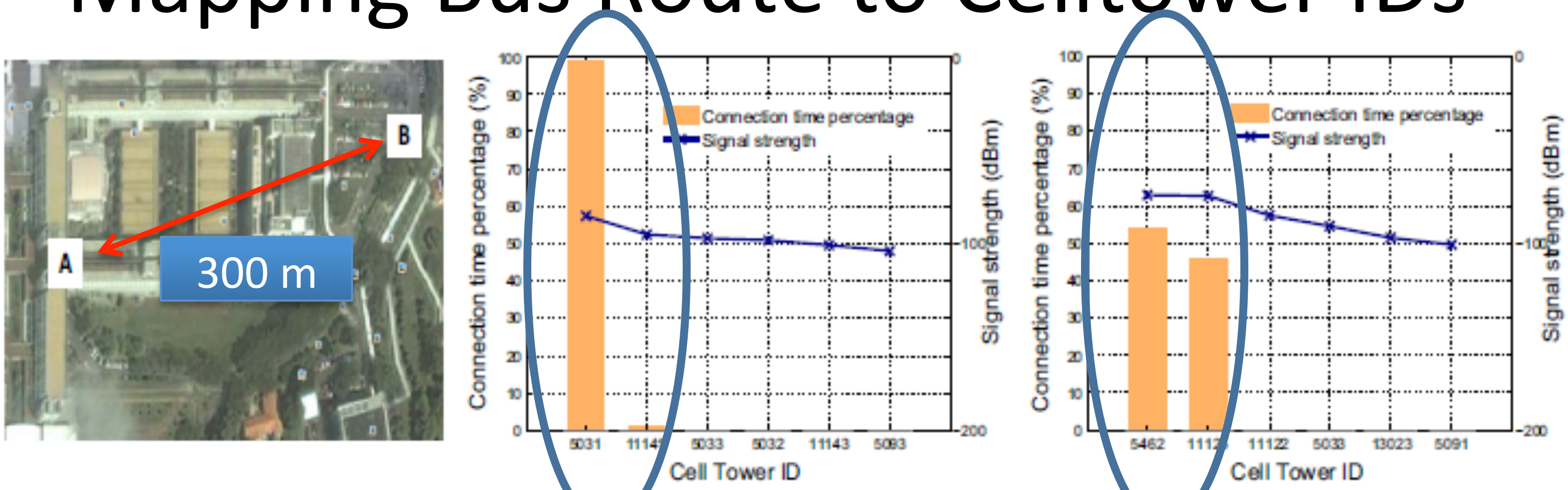
Cell tower IDs can be used to characterize the route of the bus

System Design Challenges



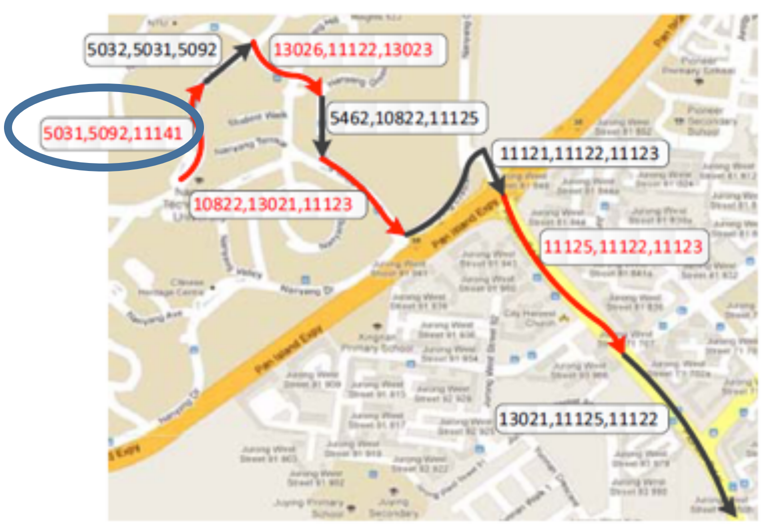
System Architecture

Mapping Bus Route to Celltower IDs



(a) Celltower coverage (b) Connection at position A (c) Connection at position B

Top-3 Strongest Celltower ->Signature for bus route segments



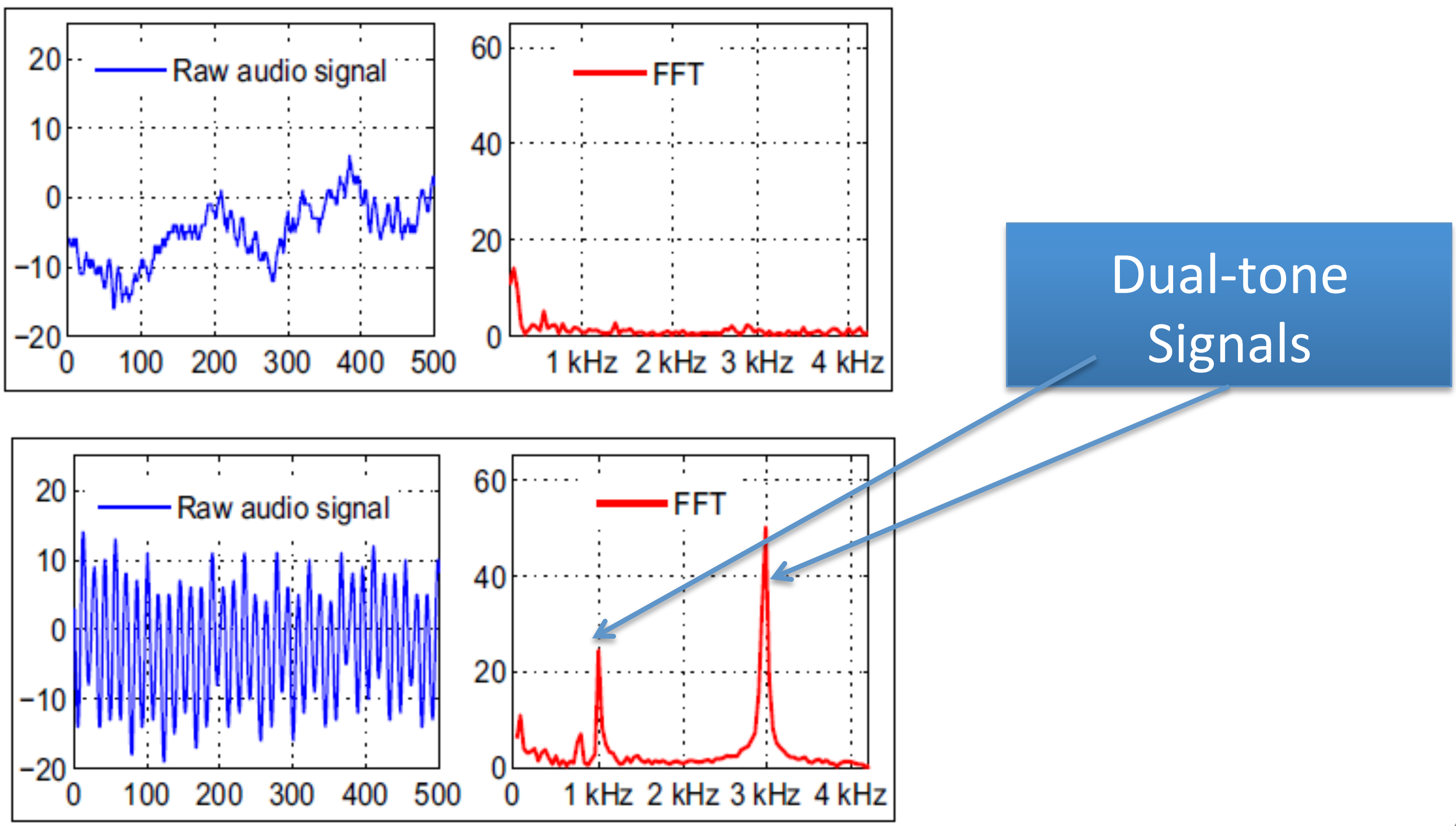
Celltower sequence along a bus route

Bus Detection

Q: How to detect whether a user is on the bus or not?

Bus Detection

Audio Detection: Short Beep Response from IC Card Reader



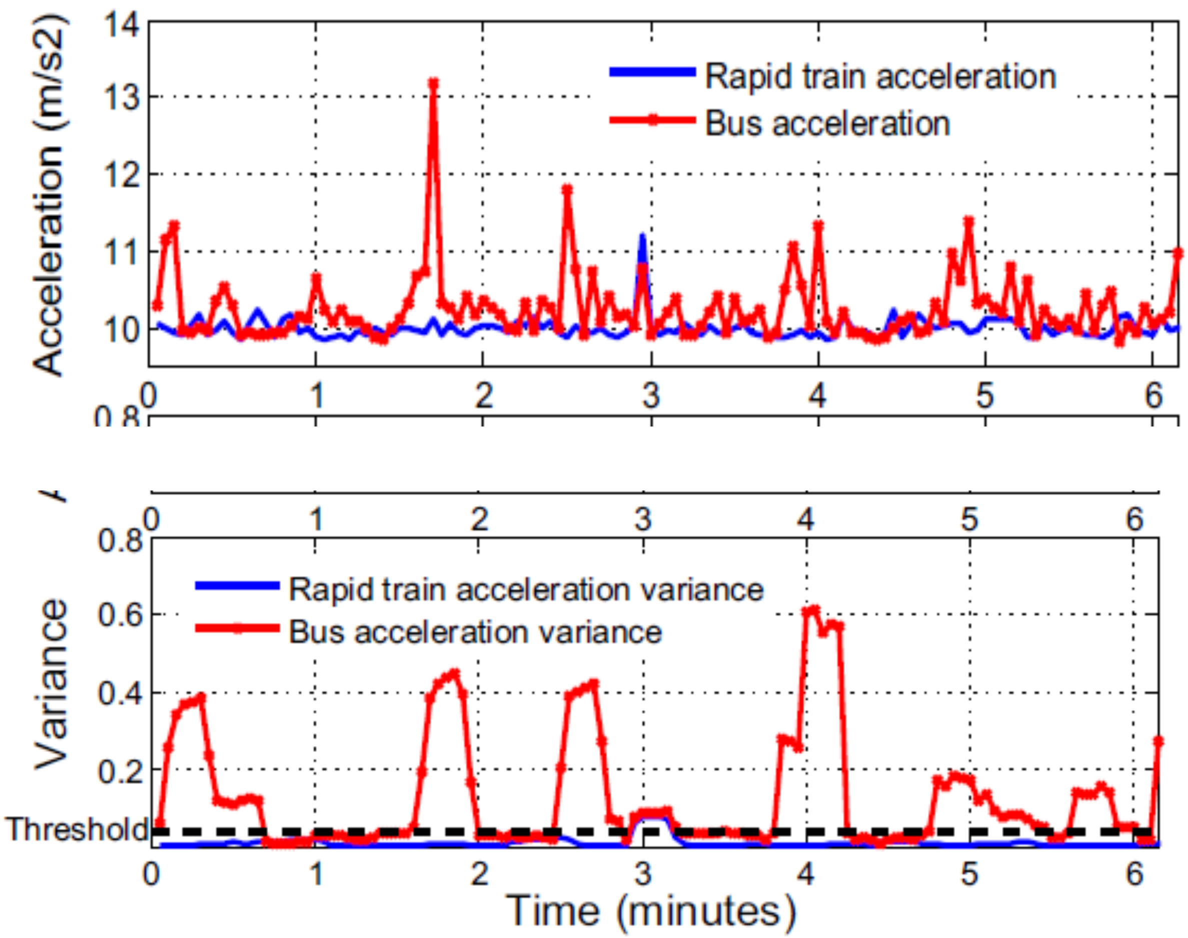
(b) IC card reader indication audio signal

Rapid train uses the same IC card system

Q: How to decide if a user gets on a bus or a rapid train?

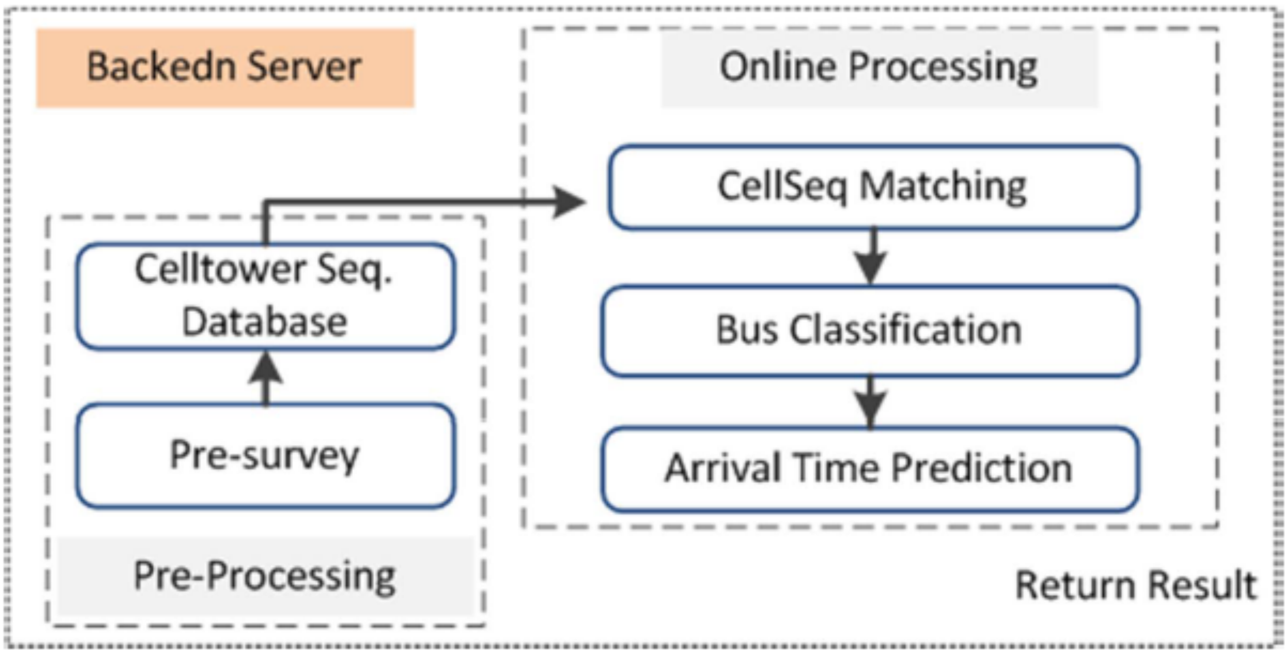
Bus Detection

Accelerometer detection: Bus vs Rapid Train



A rapid train moves at a more table speed than a bus.

Backend Server



- Pre-survey: Cell tower sequence database
- Online processing:
  - Cell tower sequence matching
  - Bus classification
  - Arrival time prediction

Bus Classification

Modified Smith-Waterman Algorithm

$$f(s_w) = 0.5^{w-1}$$

w: rank of signal strength

Find the Route with the highest matching score!

Database celltower set seq.	19 20 21	1 2 3	4 5 6	7 8 9	10 11 12	13 14 15	16 17 18	22 23 24	$\sum$
Uploaded seq.		1	–	8	10	15	16		
Score	0	+1	-0.5	+0.5	+1	+0.25	+1	0	3.25

Overlapped route

- Survey 50 bus route

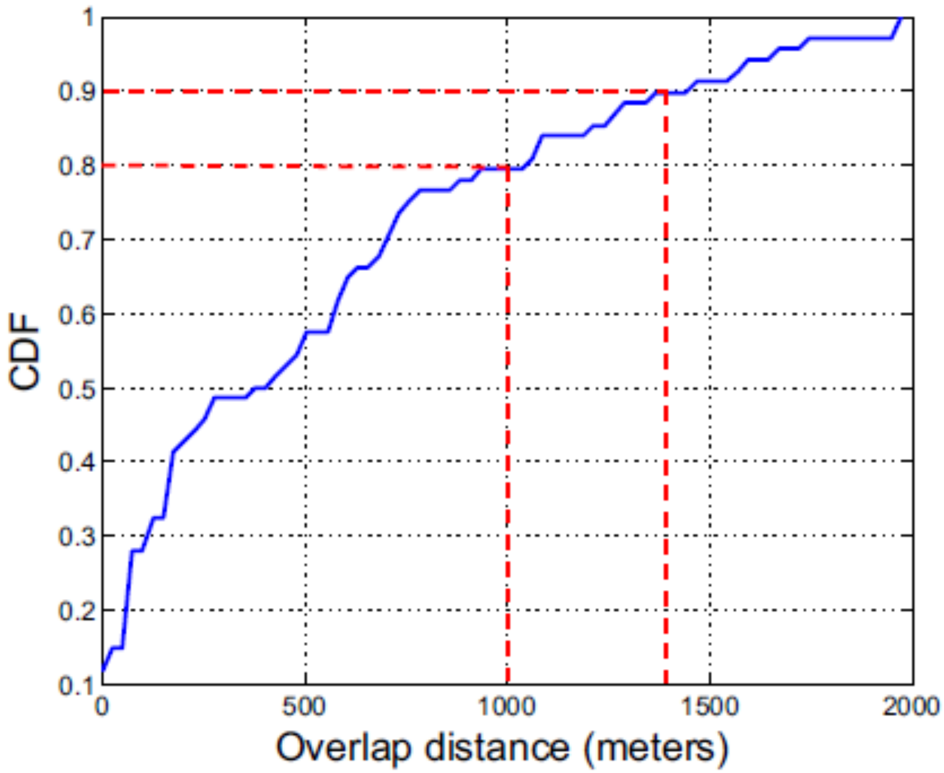
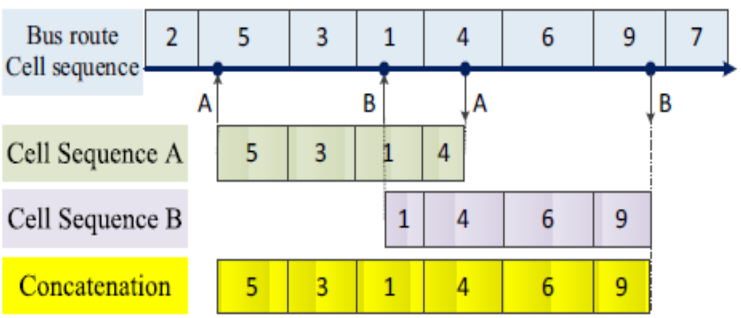


Figure 11: CDF of the overlapped route length

Range of cell tower:
300-900 meters

threshold of celltower sequence length : **7**

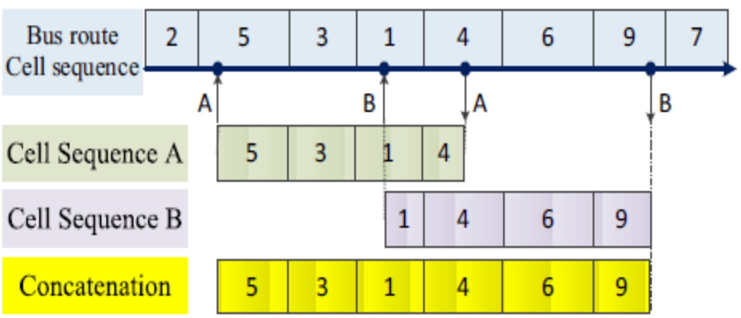
Celltower Sequence Concatenation



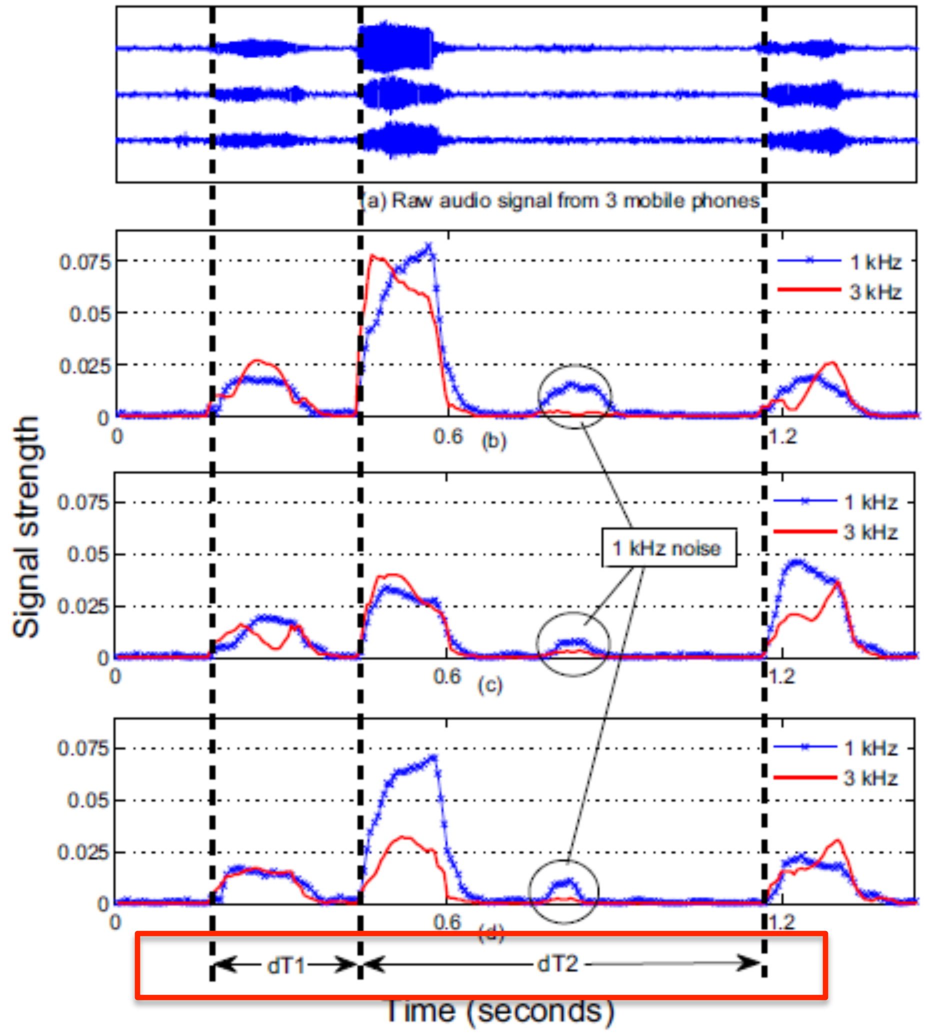
What if sequence lengths from users are too short?

Signals of 3 users on the same bus

Celltower Sequence Concatenation

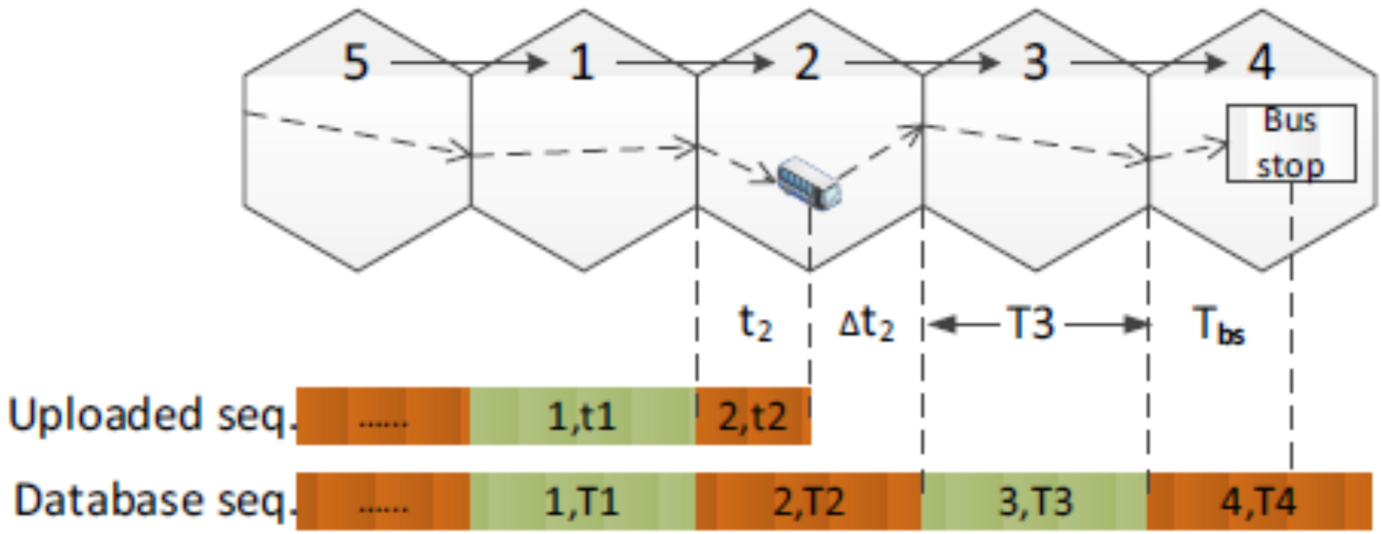


Time intervals between consecutive beep signals can fingerprint each bus in time domain



Signals of 3 users on the same bus

Arrival Time Prediction



$$T = T_2 - t_2 + T_3 + t_{bs}$$

$$T = \sum_{i=k}^{q-1} T_i - t_k + t_q$$

Evaluation : Experimental Methodology

- Mobile phones

- ✓ Accelerometer
- ✓ Microphone
- ✓ 1GB RAM
- ✓ 1.2GHz Cortex-A9 Processor

Samsung i9100

- ✓ Accelerometer
- ✓ Microphone
- ✓ 768MB RAM
- ✓ 1GHz Scorpion Processor

HTC Desire S

- Buses

SBS Transit Bus

Campus Shuttle Bus

Evaluation : Experimental Methodology

- Experiment environment
  - 4 campus shuttle bus routes
  - 2 SBS transit bus route 179 and 241

Opposite Direction!

Route	Length	Avg. vel.	Stop	Seq. Length
A	4.0km	22.1km/h	11	14-15
B	3.8km	21.2km/h	9	9-10
C	5.5km	20.6km/h	13	16-17
D	5.8km	18.3km/h	9	20-22

Table 3: Campus bus route length, average velocity, number of bus stops, and celltower sequence length

Route	A	B	C	D
A	–	1.4km	3.4km	1.9km
B	1.4km	–	2.1km	0km
C	3.4km	2.1km	–	1.9km
D	1.9km	0km	1.9km	–

Table 4: The lengths of shared bus routes

Evaluation: Bus Detection Performance

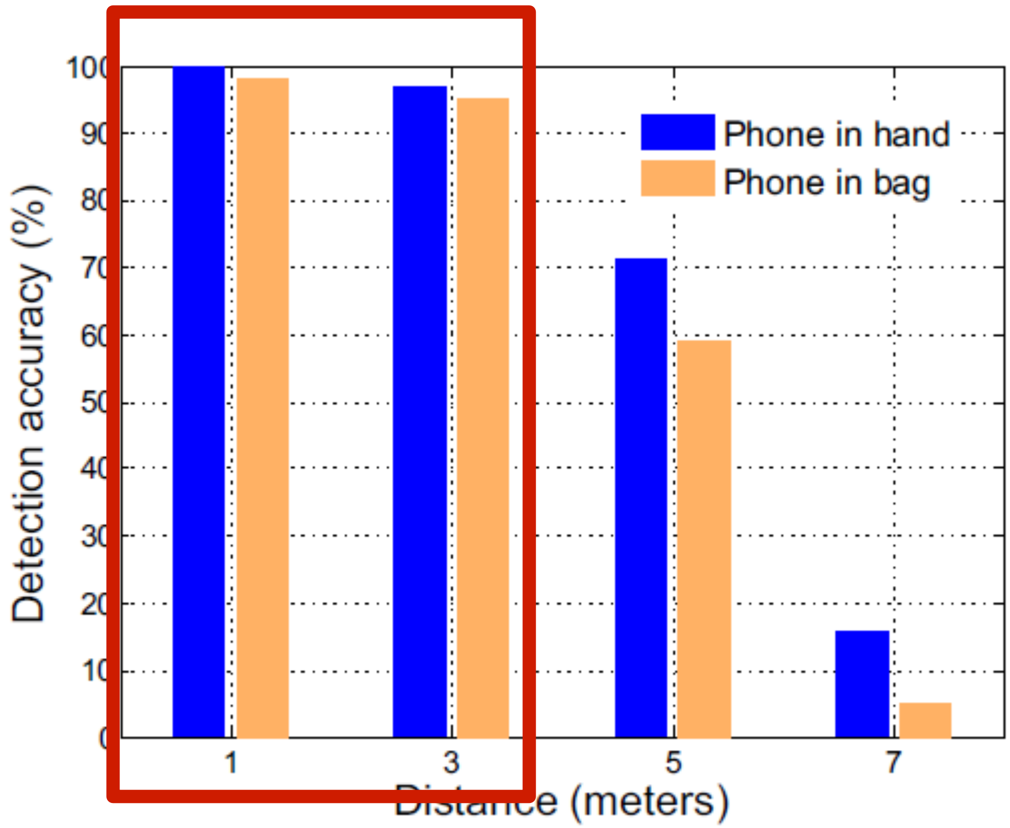


Figure 17: Bus detection accuracy

Normal Distance on Bus: 0.5 m

Evaluation: Bus vs. MRT Train

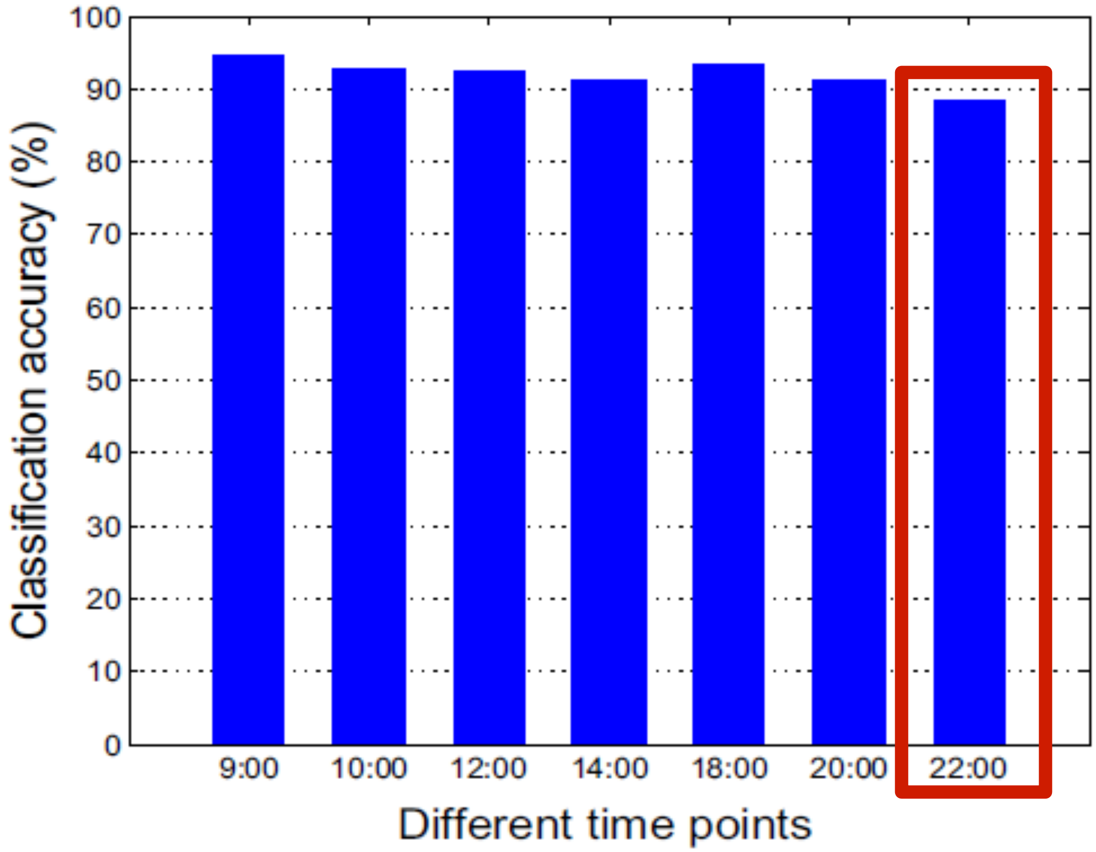
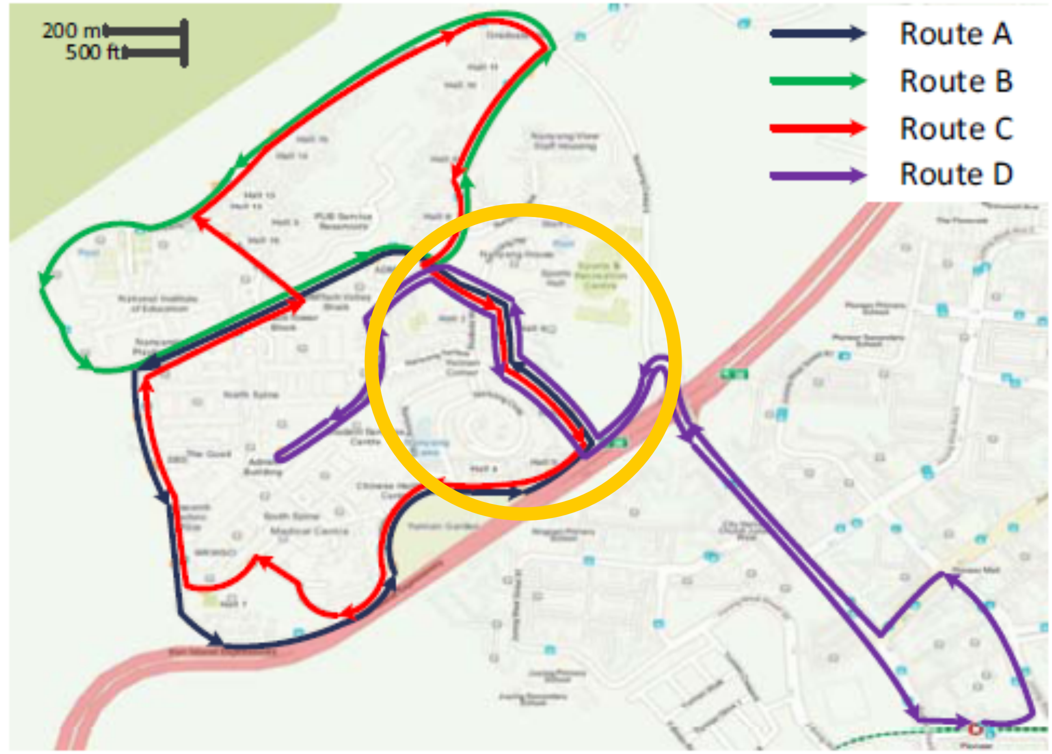


Figure 18: Bus vs. MRT using accelerometer

False detection: Driving along straight routes late during night time

Evaluation: Bus C



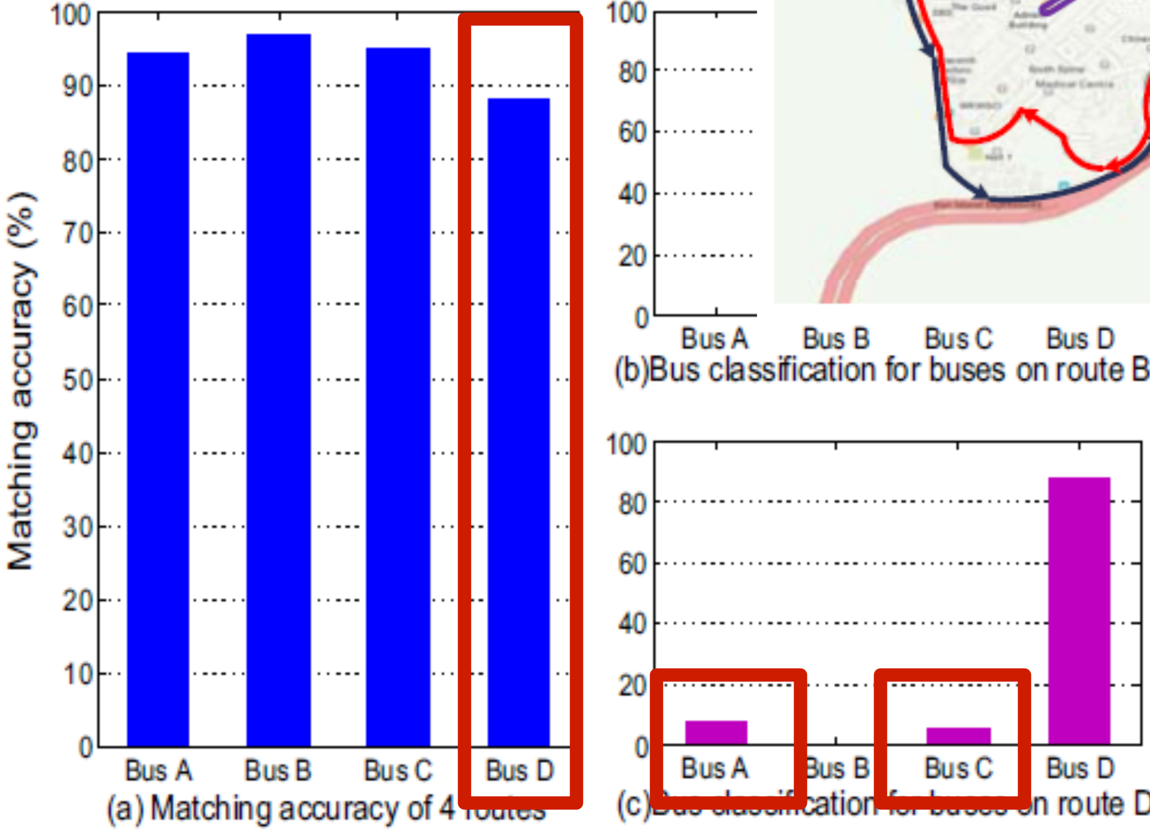
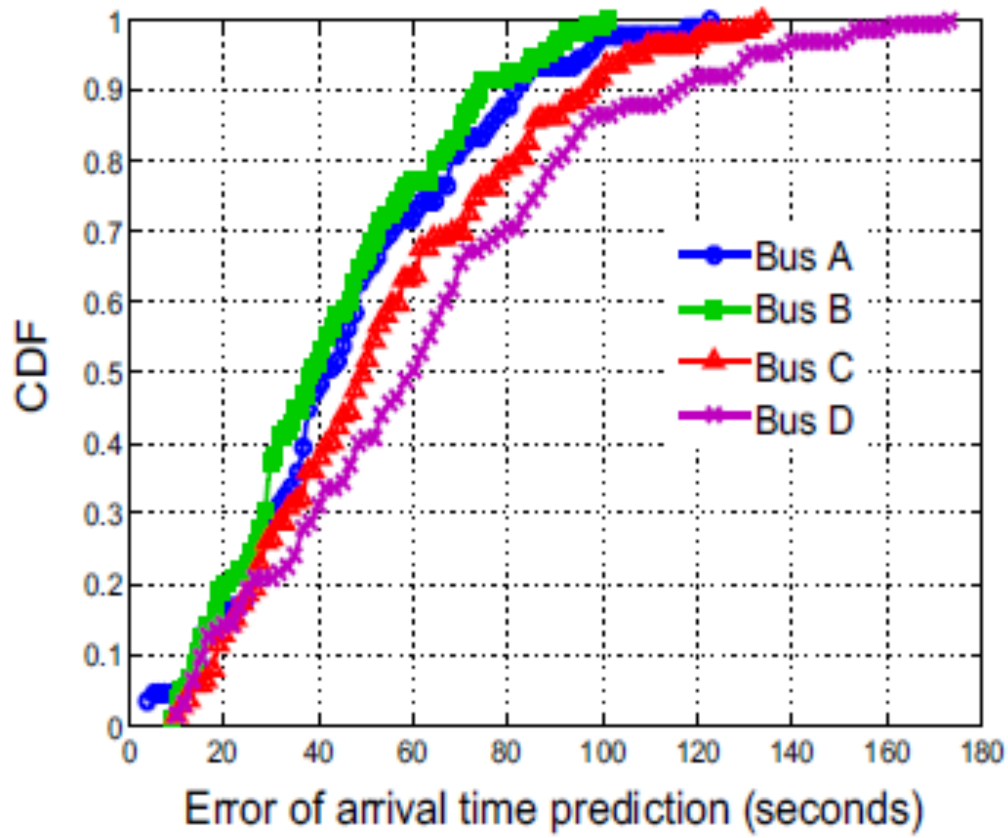


Figure 19: Bus classification accuracy

Overlapped routes are in the same direction!

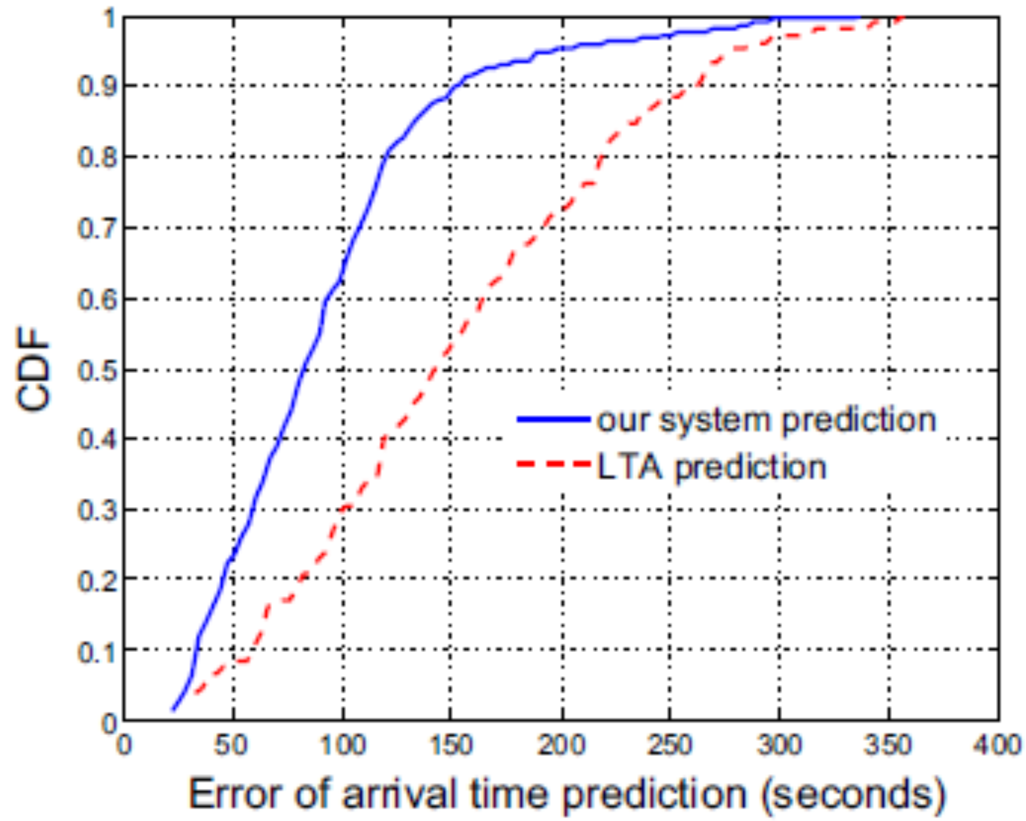
Evaluation: Arrival Time Prediction



(a) Bus arrival time prediction error

Campus Bus: median errors: 40-60 s

Evaluation: Arrival Time Prediction



(c) Our system v.s. LTA

Public Bus: median errors: 80s (this paper) vs 150s (LTA)

Evaluation: System Overhead

- Energy Consumption (Battery Life)

Sensors	Samsung i9100	HTC Desire
No sensor	18.2	15.3
Accelerometer 20Hz	18.0	15.2
Microphone 8kHz+FFT	17.5	14.9
Celltower 1Hz	17.8	15.0
GPS 1Hz	9.7	6.4

Table 6: Battery duration for different sensor settings (in hours)

What are the limitations you see?

Limitations the authors claimed

- Number of passengers
  - if no sharing users on a bus, the backend server may miss the bus
- First few bus stops
  - short celltower sequence, arrival time may not be timely updated
- Overlapped routes
  - classifying bus routes sharing substantial portion of overlapped routes remain challenging
  - use bus speed to differentiate

Future Extensions

- Preprocessing phase with crowdsourcing:
  - Querying user -> Sharing user
- Alternative reference points:
  - Roadside WiFi
- Trip planning:
  - From “how long to wait” to “where to go”

Q&A