SurroundSense: Mobile Phone Localization via Ambience Fingerprinting

CSCI780 Sensors and Ubiquitous Computing

Gang Zhou

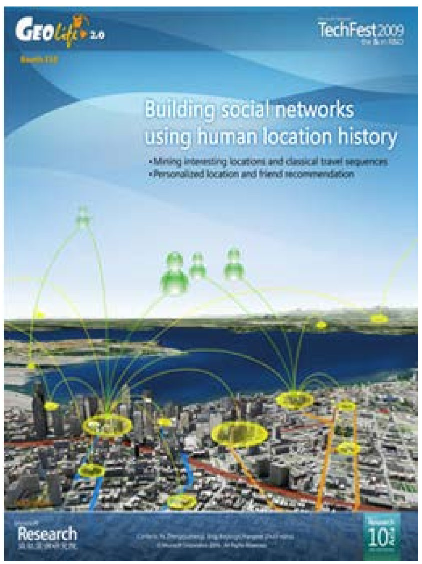
Computer Science

College of William and Mary

Based on slides from Romit Roy Choudhury

Introduction

- Location-Based Apps (LBAs):
  - **GeoLife shows grocery list when near Walmart**
  - **MicroBlog queries users at a museum**
  - **Location-based ad: Phone gets coupon at Starbucks**



- iPhone AppStore: 3000 LBAs, Android: 500 LBAs

Introduction

Most emerging location based apps

do not care about the **physical location**

GPS: Latitude, Longitude

Instead, they need the user's **logical location**

Starbucks, RadioShack,
Museum, Library

Physical vs. Logical

- Unfortunately, most existing solutions are physical

  - GPS
  - GSM based
  - SkyHook
  - Google Latitude

  - RADAR
  - Cricket
  - …

Given this rich literature,

Why not convert from

Physical to Logical Locations?

The Dividing-Wall Problem

A New Idea: SurroundSense

Hypothesis

It is possible to localize phones by sensing the ambience

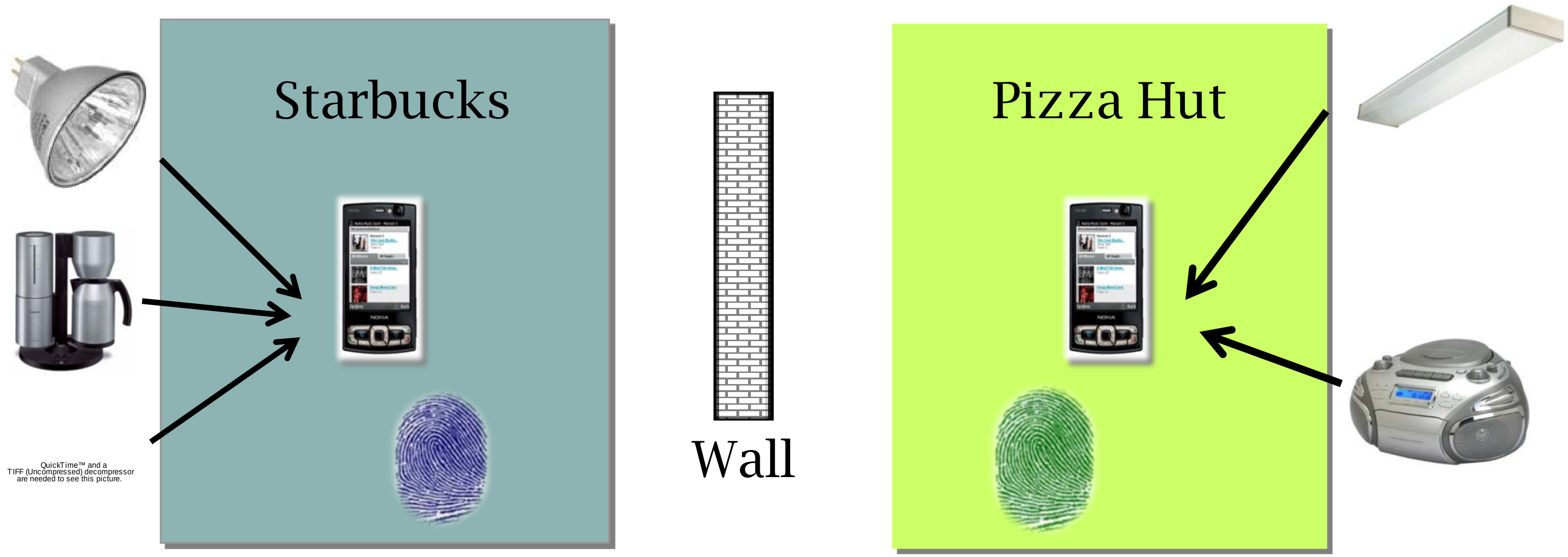
such as sound, light, color, movement, WiFi …

Outline

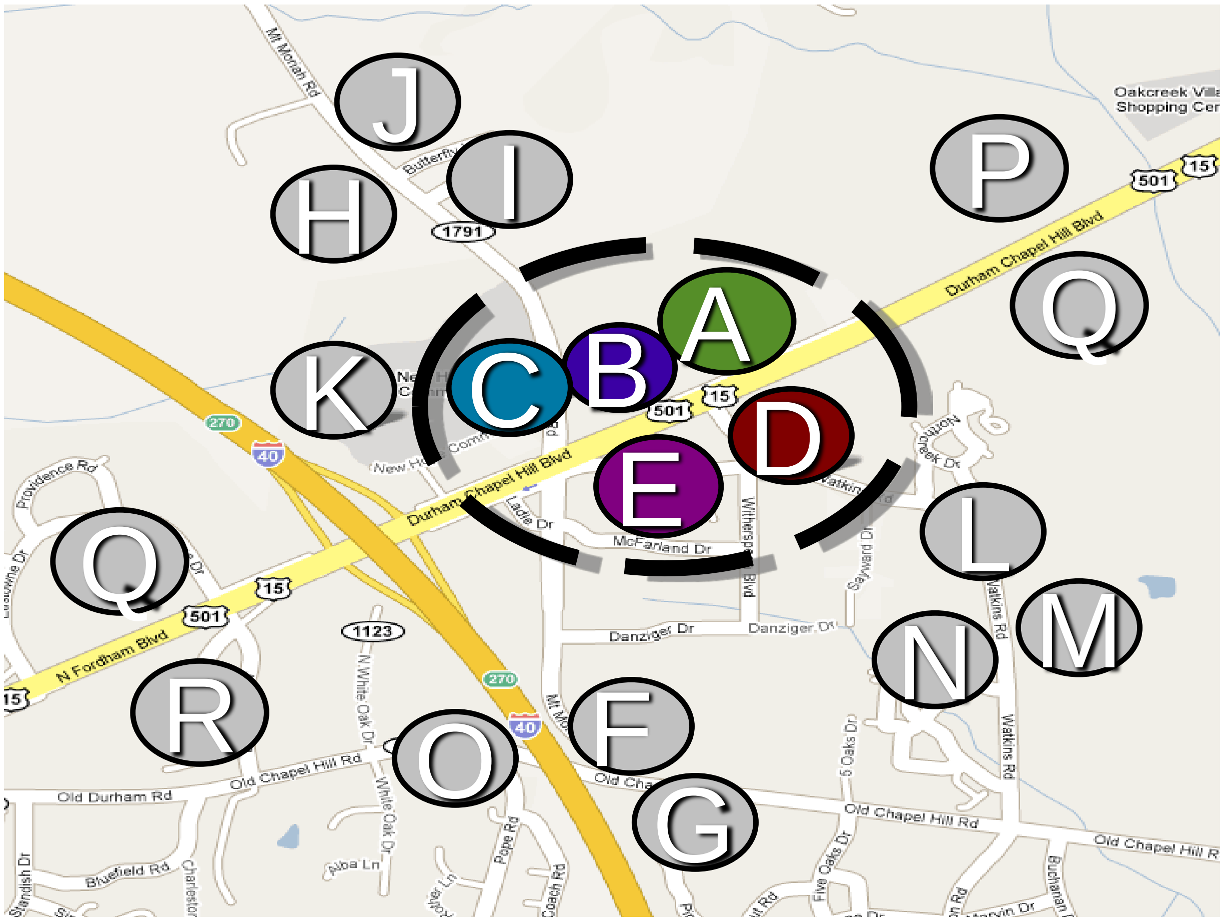
- Introduction
- The Main Idea
- Feasibility and Potential
- System Architecture
- Fingerprinting Details
- Discussions
- Performance Evaluation
- Conclusions and Future work

SurroundSense

- Multi-dimensional fingerprint
  - **Based on ambient sound/light/color/movement/WiFi**



GSM provides macro location (strip mall)
SurroundSense refines to Starbucks



Feasibility and Potential

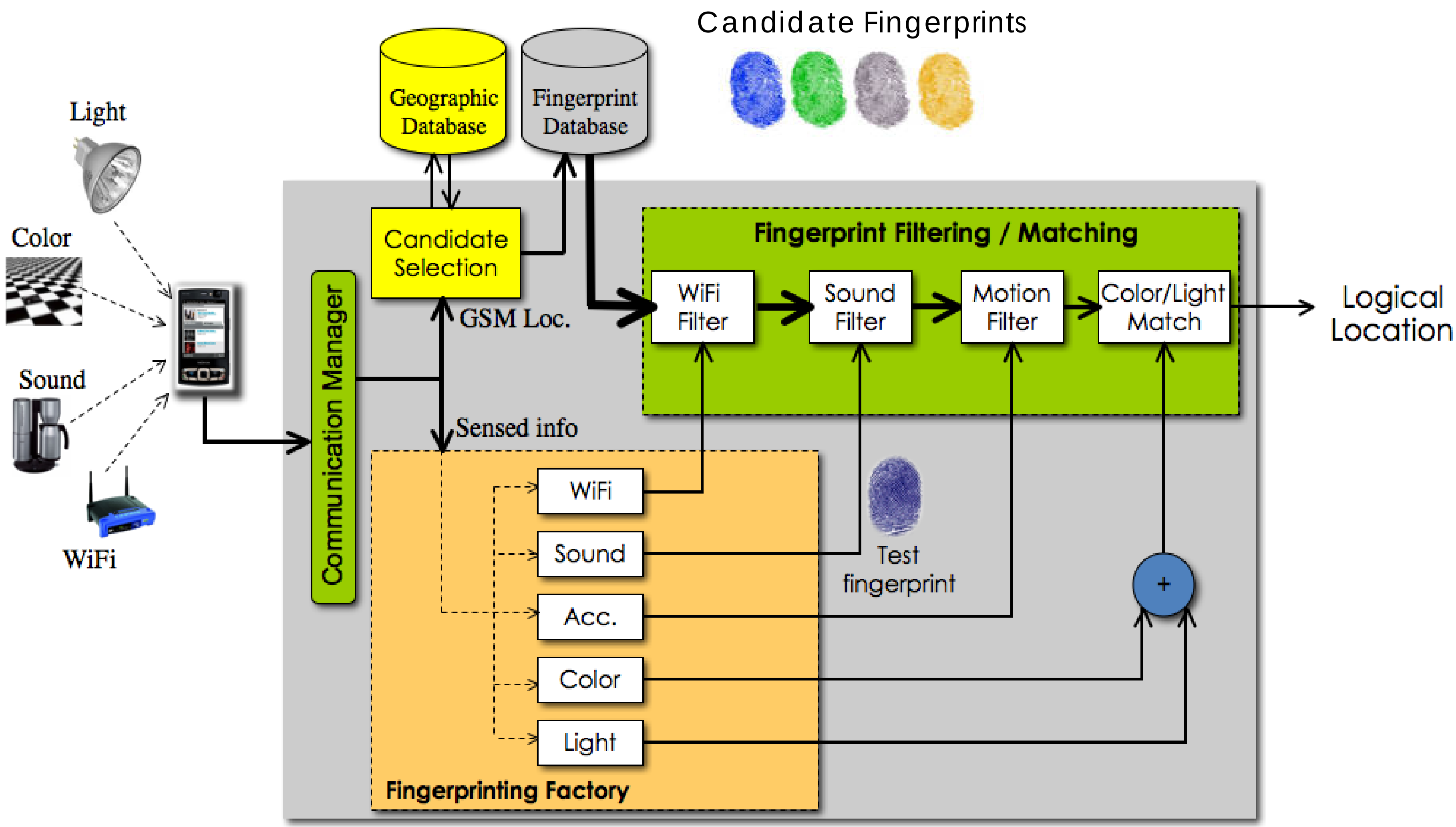
The Intuition:

Economics forces nearby businesses to be diverse

Not profitable to have 3 adjacent coffee shops with same lighting, music, color, layout, etc.

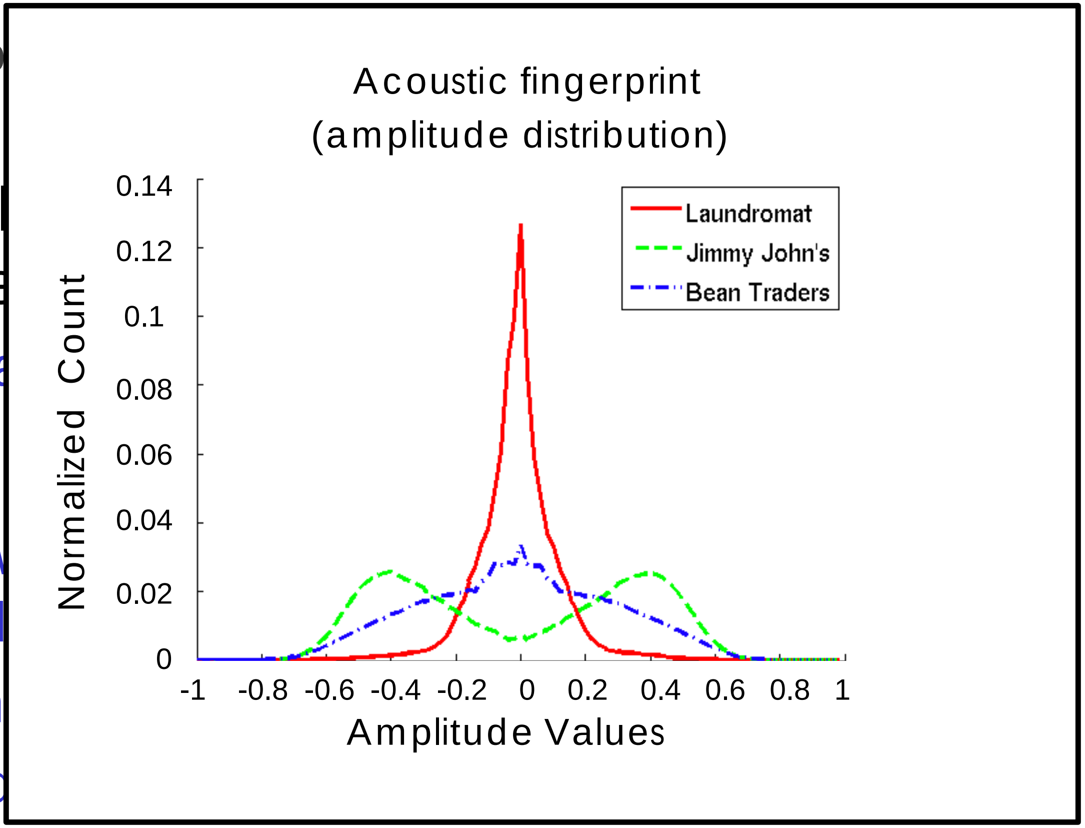
SurroundSense exploits this ambience diversity

Architecture: Filtering & Matching



Fingerprinting Sound

- Fingerprint generation : Signal amplitude
  - Amplitude values divided in 100 equal intervals
  - Sound Fingerprint = 100 normalized values
    - $value_X$ = # o
- Filter Metric: Eucl
  space between te
  - Discard candida
- Threshold Γ
  - Compute pair-w
    fingerprints at d
  - d_i = 95th percen
  - Γ = maximum d



Fingerprinting Motion

- Want to recognize
  - Sitting (restaurants, cafes, haircutters)
  - Slow Browsing (bookstores, music stores, clothing)
  - Speed-Walking (groceries)
- Process
  - 4 samples per second for 3-axes accelerometer
  - Moving average of window 10
  - SVM classification into “stationary” and “motion” with features of “mean” and “variance”
  - Filter Metric:

$$R = \frac{t_{moving}}{t_{static}} \quad \begin{cases} 0.0 \le R \le 0.2 & \text{sitting} \\ 0.2 \le R \le 2.0 & \text{slow browsing} \\ 2.0 \le R \le \infty & \text{speed-walking} \end{cases}$$

    - Discard candidate fingerprints with different classification

Fingerprinting Motion

Queuing

Seated

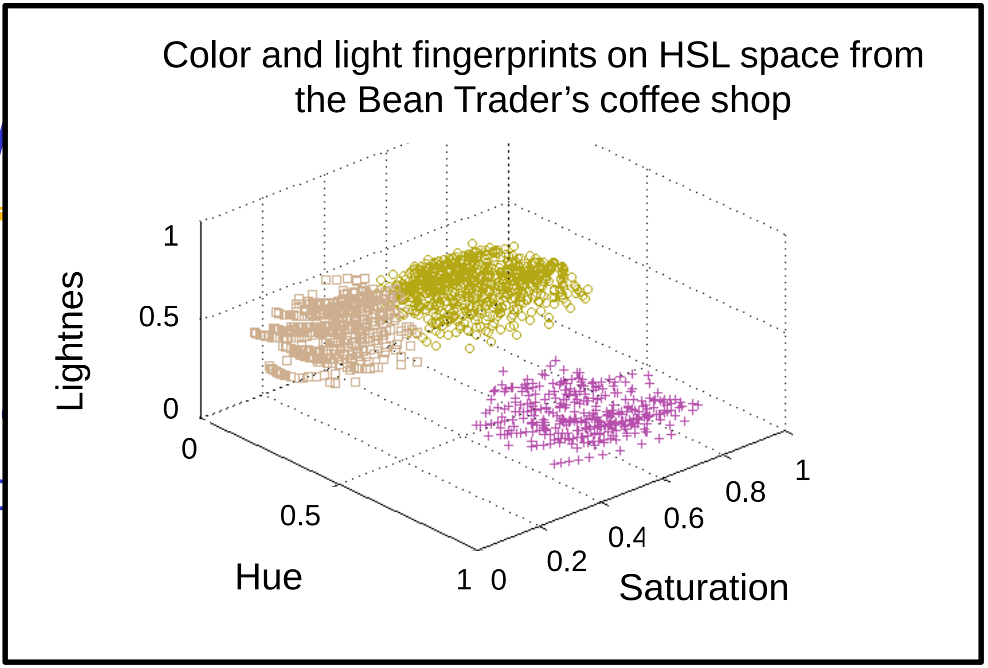
Pause for product browsing

Walk more

Short walks between product browsing

Quicker stops

Fingerprinting Color/



- Floor Pictures
  - Rich diversity across diff
  - Uniformity at the same lo
- Fingerprint generation: pictures in Hue-Saturation-Lightness (HSL) space
  - K-means clustering algorithm to get multiple clusters
  - Get each cluster's centroid and size
- Ranking metric

$$S_{12} = \sum_{i,j} \frac{1}{\delta(i,j)} \frac{SizeOf(C_{1i})}{T_1} \frac{SizeOf(C_{2j})}{T_2}$$

Fingerprinting WiFi

- Matching or filtering?
  - Used as matching in the absence of light/color
  - Otherwise, used as a filter
- Detailed method
  - Record AP MAC addr. from received beacons every 5s
  - Compute the fraction of times each unique MAC address m was seen over all recordings: f(m)
  - A tuple of fractions forms the WiFi fingerprints
  - Intuition: get a large S when a MAC address occurs frequently in both f1 and f2

$$S = \sum_{m \in M} (f_1(m) + f_2(m)) \frac{min(f_1(m), f_2(m))}{max(f_1(m), f_2(m))}$$

Evaluation Methodology

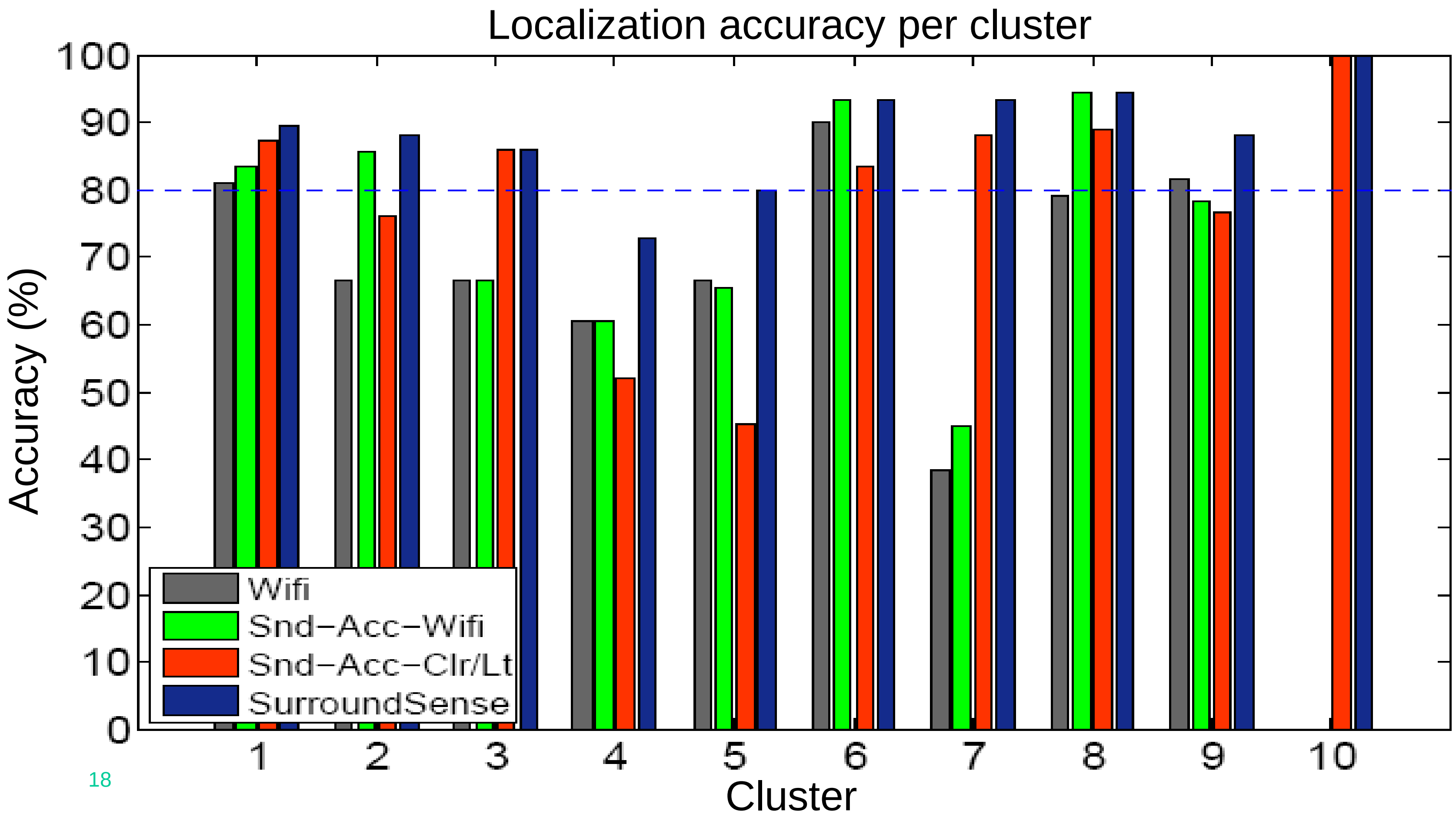
- 51 business locations
  - **46 in Durham, NC**
  - **5 in India**

- Data collected by 4 people
  - **12 tests per location**

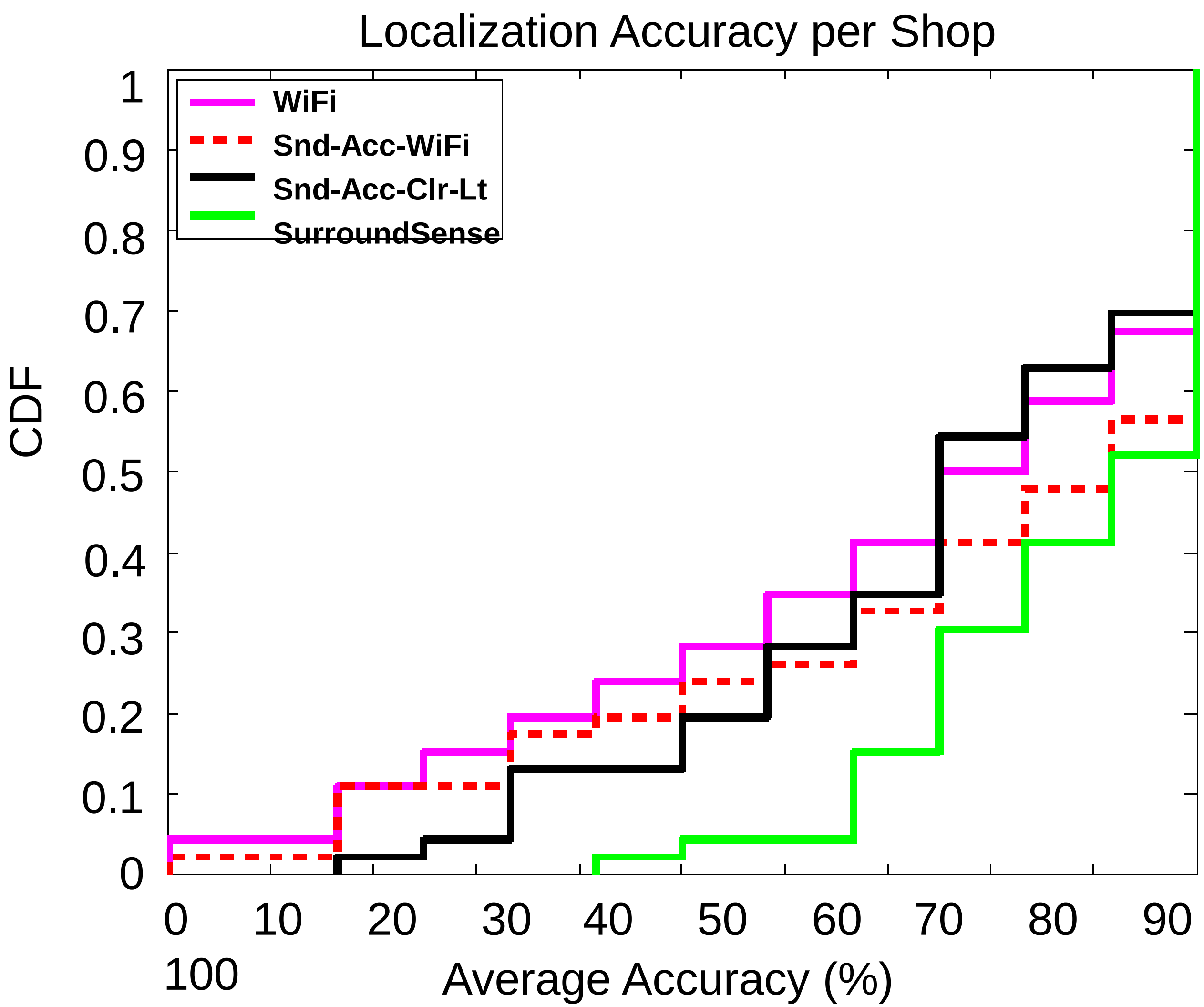
- Mimicked customer behavior

Per-Cluster Accuracy

Cluster	1	2	3	4	5	6	7	8	9	10
No. of Shops	4	7	3	7	4	5	5	6	5	5



Per-Shop Accuracy

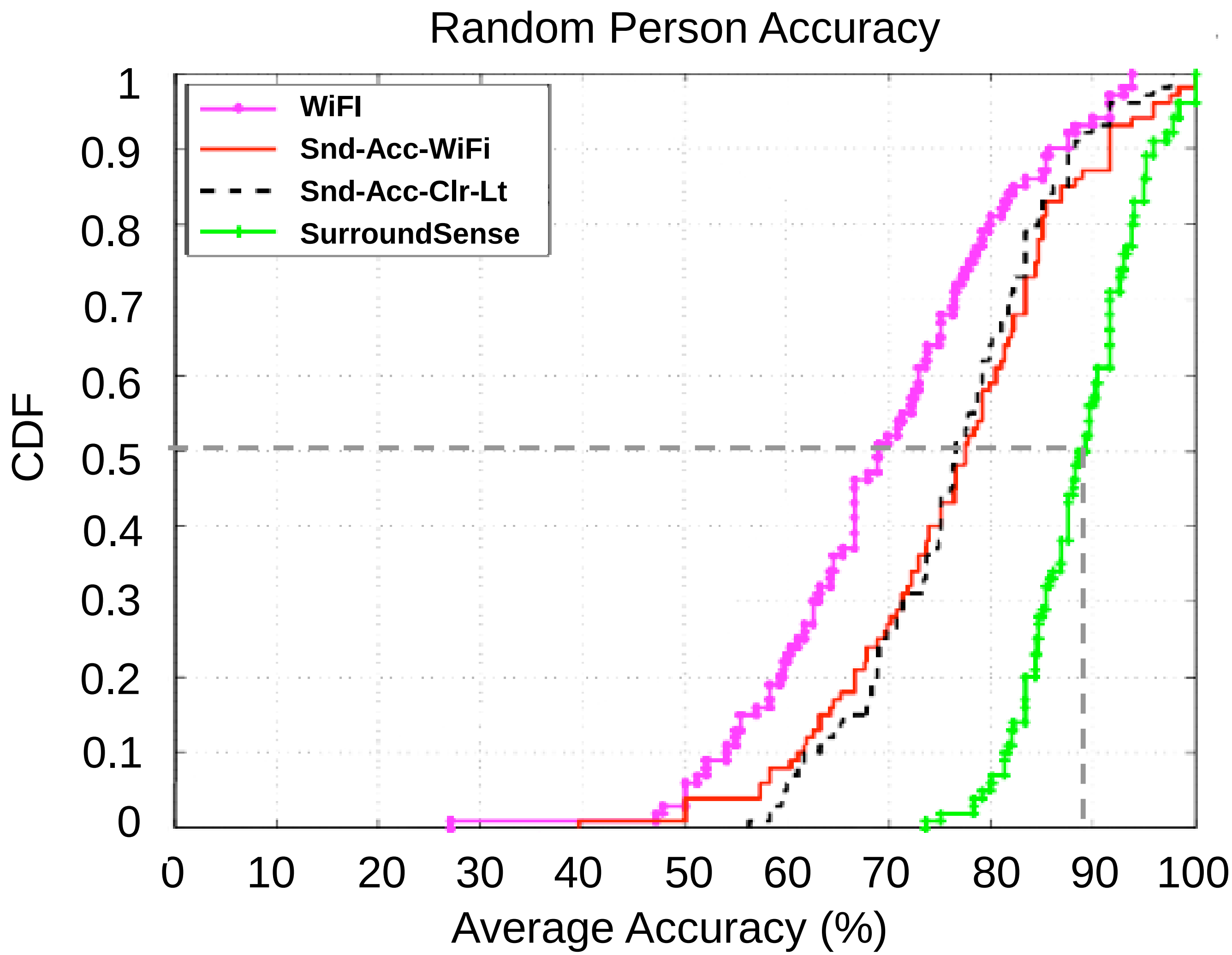


Per-Scheme Accuracy

Mode	WiFi	Snd-Acc-WiFi	Snd-Acc-Lt-Clr	SS
Accuracy	70%	74%	76%	87%

Average accuracy across clusters 1-9
(WiFi not available in cluster 10

Per-User Accuracy



Conclusions

- Identifying the possibility of fingerprinting a logical location based on ambient sound, light, color, and human movement
- An experimental framework that creates a fingerprint database and performs fingerprint matching for test samples
- Evaluation of the scheme in business locations in a university town

Discussion

- How to deal with time varying ambience?
  - **Collect ambience fingerprints over different time windows**
- What if phones are in pockets?
  - **Use sound/WiFi/movement**
  - **Opportunistically take pictures, need to**
    - Detect phone when out of pocket
    - Takes pictures when camera pointing downward
- How to populate fingerprint database?
  - **War-sensing**

Discussion (cont.)

- How to make it energy efficient, since continuous sensing likely to have a large energy draw?
  - Duty cycling sampling, computation, etc?
- How to do localization in real-time, since user's movement requires time to converge?
  - Find features that need less sensor samples?
- Is it promising to use this idea in non-business locations, since, e.g., different houses may have the same style?
  - Need to think hard about features…probably more human-centered features…