



Universität Stuttgart
Institute of Parallel and
Distributed Systems (IPVS)
Universitätsstraße 38
D-70569 Stuttgart

Public Sensing – Using Your Mobile Phone for Crowd Sourcing

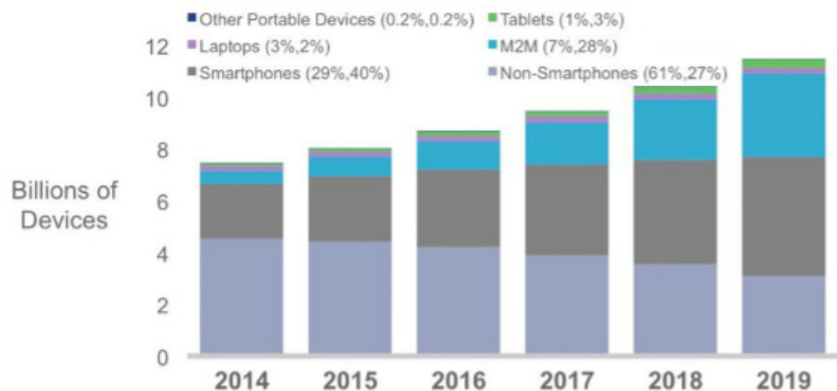
55th Photogrammetric Week

September 10, 2015

Stuttgart, Germany

Kurt Rothermel

Increasing “Density” of Mobile Devices



Figures in parentheses refer to 2014, 2019 device share.
Source: Cisco VNI Mobile, 2015

11.4 Billion Mobile Devices by 2019

- 8.2 billion personal devices
- 3.2 billion M2M (GPS systems in cars, asset tracking, medical apps, ...)



Research Group
“Distributed Systems”

Universität Stuttgart
IPVS

Increasing “Density” of Mobile Devices

Announcement of
Pope Benedict XVI



Announcement of
Pope Francis



Mobile Devices Become Powerful Sensor Platforms



[source: Microsoft]

- Optical heart rate monitor
- Accelerometer, gyrometer
- GPS
- Microphone
- Ambient Light Sensor
- Galvanic Skin Response (GSR) sensors
- UV sensor



[source: Google Project Tango]

- NVIDIA Tegra K1 processor
 - ARM Cortex A15 “big.LITTLE” architecture
 - 4 cores @ 2.3 GHz + 1 Core
 - 192 CUDA cores
- 4 GB RAM, 128 GB SSD
- Motion tracking camera
- 4 MP camera
- 3D depth sensing
- Barometer
- Compass
- GPS
- Gyroscope
- Ambient light

Vision: Public Sensing

Crowd-sense data using mobile devices

- Good coverage where it matters
 - Need data where there are people
- No up-front investments for dedicated sensor network
 - No need to purchase, install, and maintain cameras, road loop sensors, etc.



IPVS

Research Group
"Distributed Systems"

5

Universität Stuttgart
IPVS

Applications: Collaborative Mapping

OpenStreetMap (OSM)

- Users collect GPS traces, manually transformed into maps
- As of August 2015:
 - 4.8 billion GPS points uploaded
 - ~ 23.000 – 31.000 active contributors each month creating nodes

Com’N’Sense project @ University of Stuttgart (ifp, IPVS)

- Users collect mobility traces indoors
- Automatically build 2.5d indoor maps and 3d indoor models
... more about this later



IPVS

Research Group
"Distributed Systems"

6

Universität Stuttgart
IPVS

Applications: Real-time Traffic Monitoring



- Collects anonymous traces from 80 Mio TomTom devices
- 5 trillion anonymously-collected data points
- 6 billion new records each day



- Users contribute route information (GPS traces) online
- Users manually report events, i.e., accidents, traffic jams, speed traps
- 50 Million users 2013 (then acquired by Google for 1.1B \$)



Research Group
"Distributed Systems"

7

Universität Stuttgart
IPVS

Applications: Urban Planning & Environmental Monitoring

Road condition monitoring:

Street Bump Project of Boston's Mayor's Office

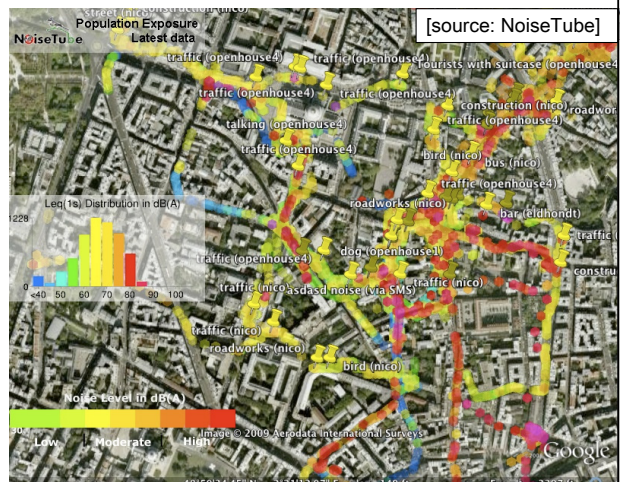
- Users collect GPS and acceleration data with car-mounted smartphones
- Data processed to detect road condition (position of potholes)

Noise pollution monitoring

- Users sense noise level with their smartphones

Air pollution monitoring

- Portable devices measuring ozone, CO, ...



Overview

- Applications
- Challenges
- Some Solutions
 - Model-based Sensing
 - Com’N’Sense Project
- Summary



IPVS

Research Group
“Distributed Systems”

9

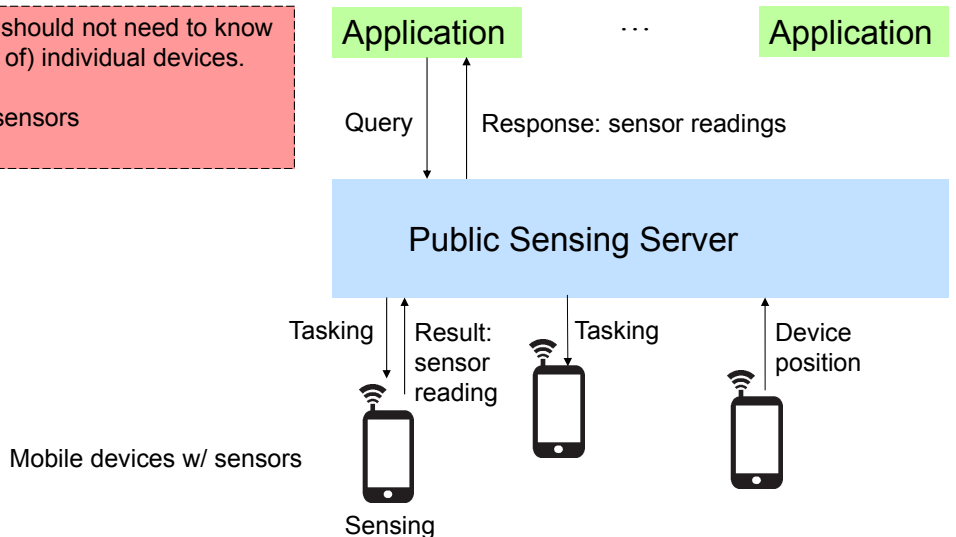
Universität Stuttgart
IPVS

Public Sensing Challenges (1)

Hiding complexity: Developing Apps should be simple

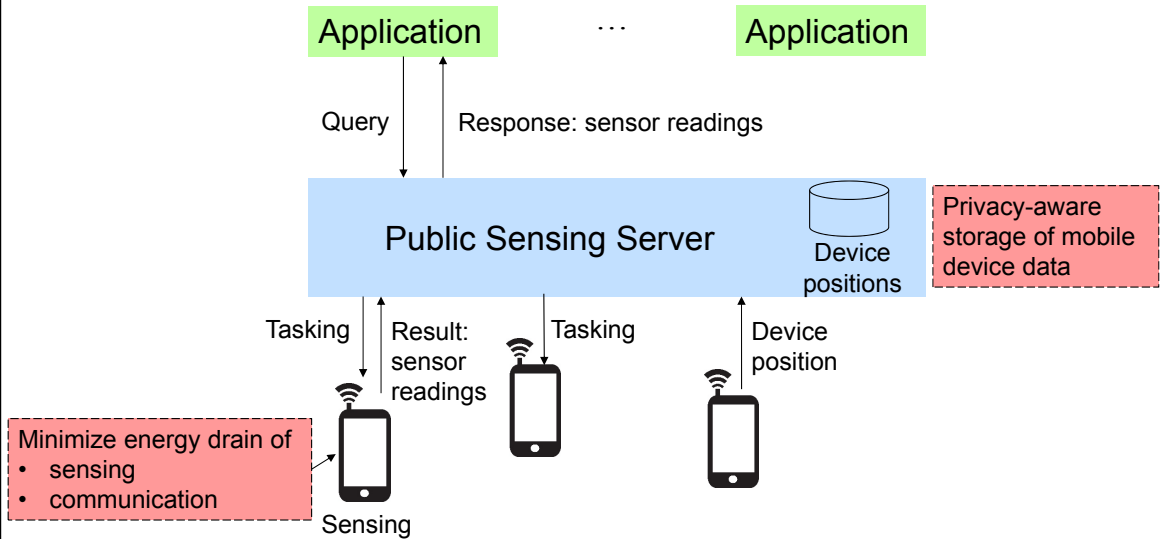
Application should not need to know (the billions of) individual devices.

- Location
- Type of sensors
- ...



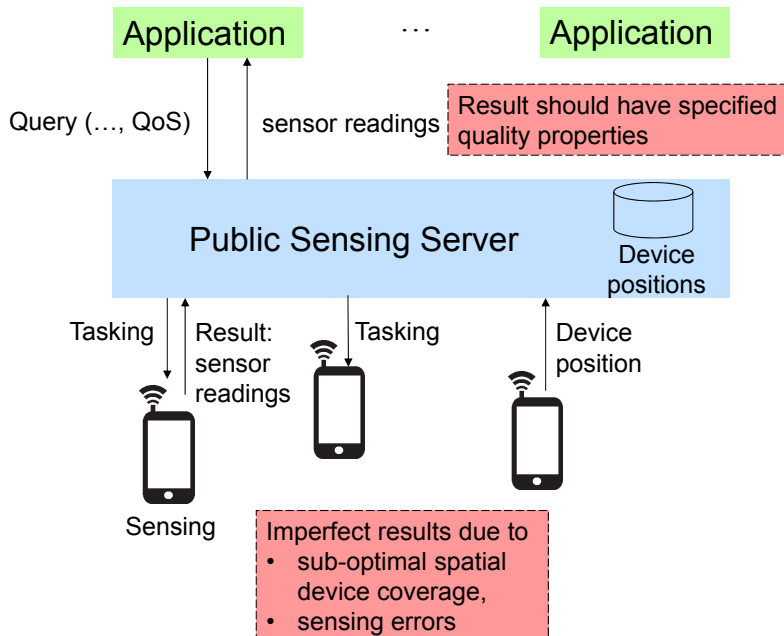
Public Sensing Challenges (2)

User acceptability: Users must be willing to participate



Public Sensing Challenges (3)

Quality of sensing: Data quality may vary substantially



Some Solutions

- **Virtual sensor abstraction**
 - Simple API – transparency of device population and mobility
- **Model-driven sensing**
 - Increase sensing quality
 - Reduce sensing and communication energy
- **Energy-efficient communication of sensor values**
 - Leveraging knowledge about mobile communication technology
 - Not covered in this talk (see <http://www.comnsense.de>)
- **Privacy-aware management of location information**
 - Storing private position data on non-trusted servers (in the “Cloud“)
 - Not covered in this talk (see <http://www.priloc.de>)



IPVS

Research Group
“Distributed Systems”

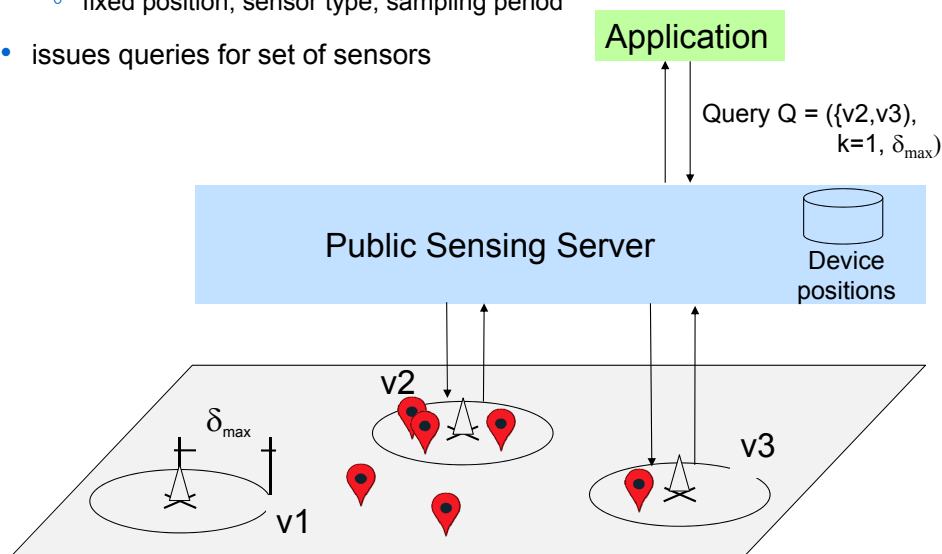
13

Universität Stuttgart
IPVS

Virtual Sensor Abstraction

Application

- defines set of virtual sensors
 - fixed position, sensor type, sampling period
- issues queries for set of sensors



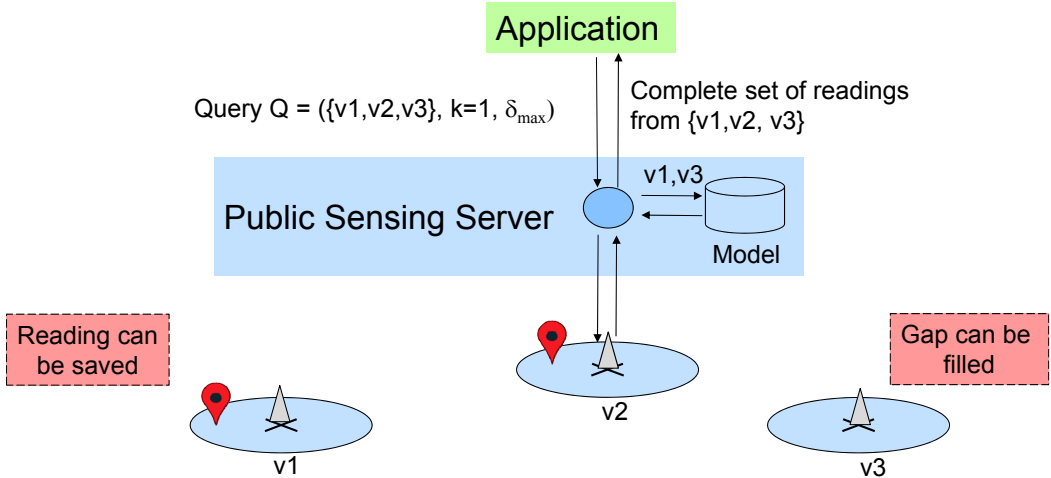
Model-driven Sensing

Model of the physical phenomenon needed

- correlation between sensor values of different virtual sensors

Model can be used to predict sensor values

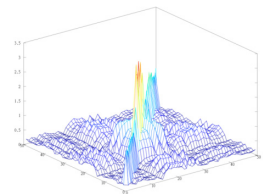
- ... instead of querying physical devices → saves energy
- ... also at places where no device is located → fill white spots



Models for Model-driven Sensing

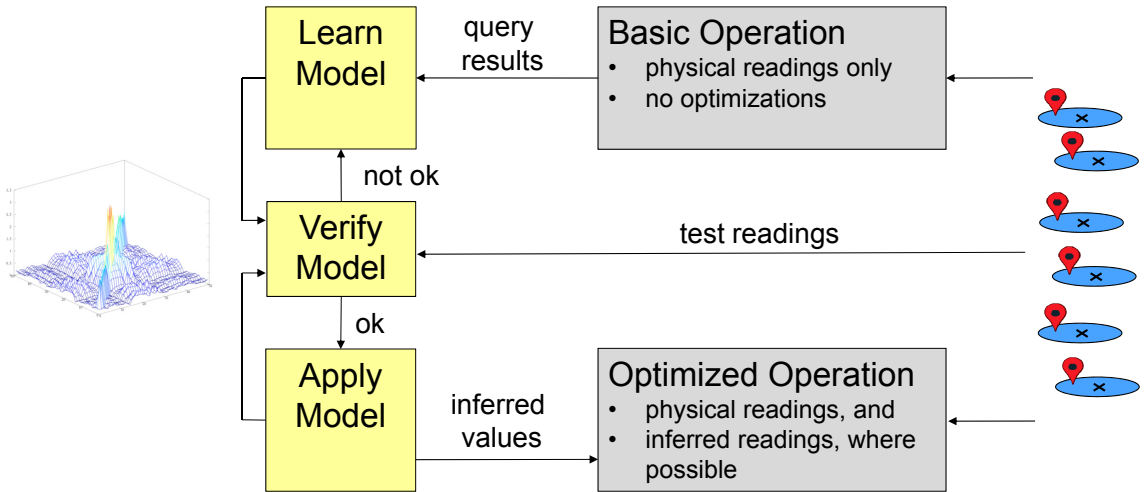
Appropriate Model depends on application

- Predicting scalar sensor values (e.g., temperature, noise, light, ...)
 - Multi-variate Gaussian Distribution (MGD)
 - Regression, autoregression, Kalman filters, ...
- Generating indoor floor plans
 - Indoor grammars



• $R_i^{room} : Space \rightarrow r_i$ Space for $i \in$ room classes found in the building
 • $R_n^{room} : Space \rightarrow r_1, \dots, r_k$ Space for $n \in$ room units found in the building
 • $R^e : Space \rightarrow \epsilon$

Online Learning the Model



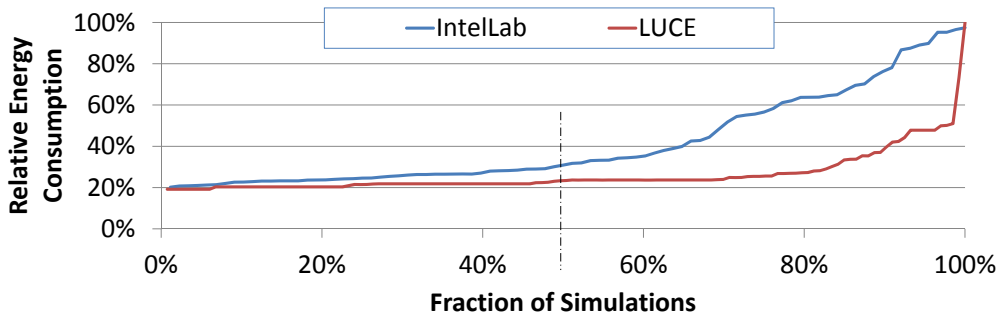
IPVS

Research Group
"Distributed Systems"

17

Universität Stuttgart
IPVS

Evaluation



Setup

- Simulation with two real-world data sets (IntelLab / LUCE)
- Sensing of temperature readings
- 1°C error limit
- 200/400 mobile nodes

Energy savings: Model-based vs non-optimized sensing

- **59 % (IntelLab data set) and 74 % (LUCE data set) energy on average**
- Fewer and smaller messages
- Energy for temperature sensing negligible



IPVS

Research Group
"Distributed Systems"

18

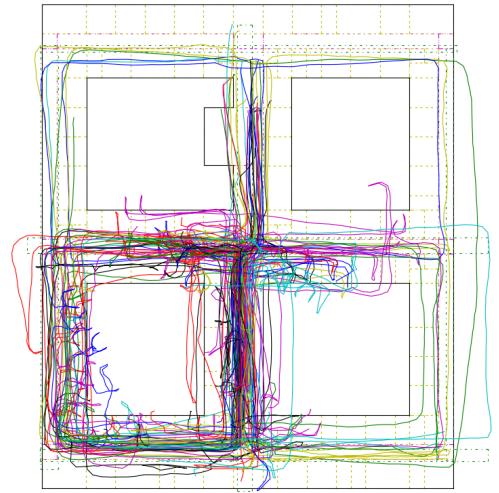
Universität Stuttgart
IPVS

Goal: Automatically create indoor maps from mobility traces

- Collect traces from crowds of users
- Infer indoor floor plan and room layout automatically from traces

Problems:

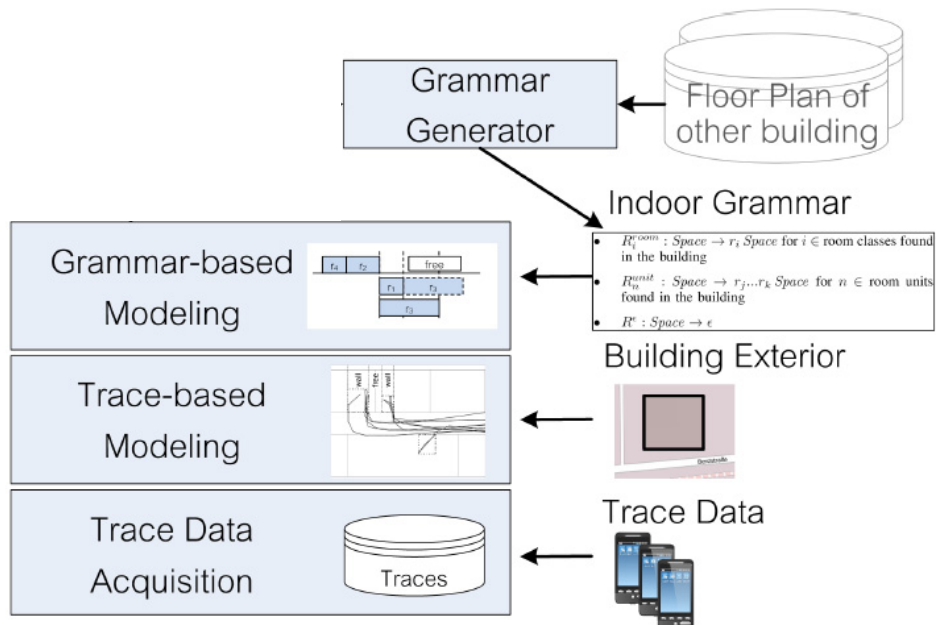
- Poor trace coverage of some areas
 - Many rooms **cannot be mapped**
- Traces are often noisy
 - Inferred indoor map is **inaccurate**



Model-based Sensing Approach:

- Use **indoor grammars** to increase quality (coverage and accuracy)

System Overview



Trace Data Acquisition with Public Sensing

- **Inertial positioning system**

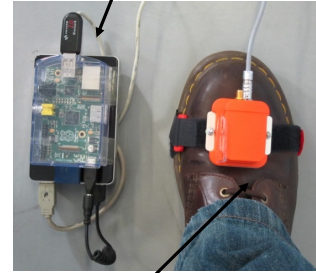
- Foot-mounted Inertial Measurement Unit (IMU)
- **Zero-velocity update** to reduce drift error
 - Still there are significant errors!

- Traces are collected **opportunistically**

- Users walk their everyday paths
- No direction or orders given to user

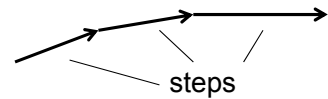
- Traces are **uploaded to server** for processing

Raspberry Pi +
Bluetooth
communication with
mobile phone

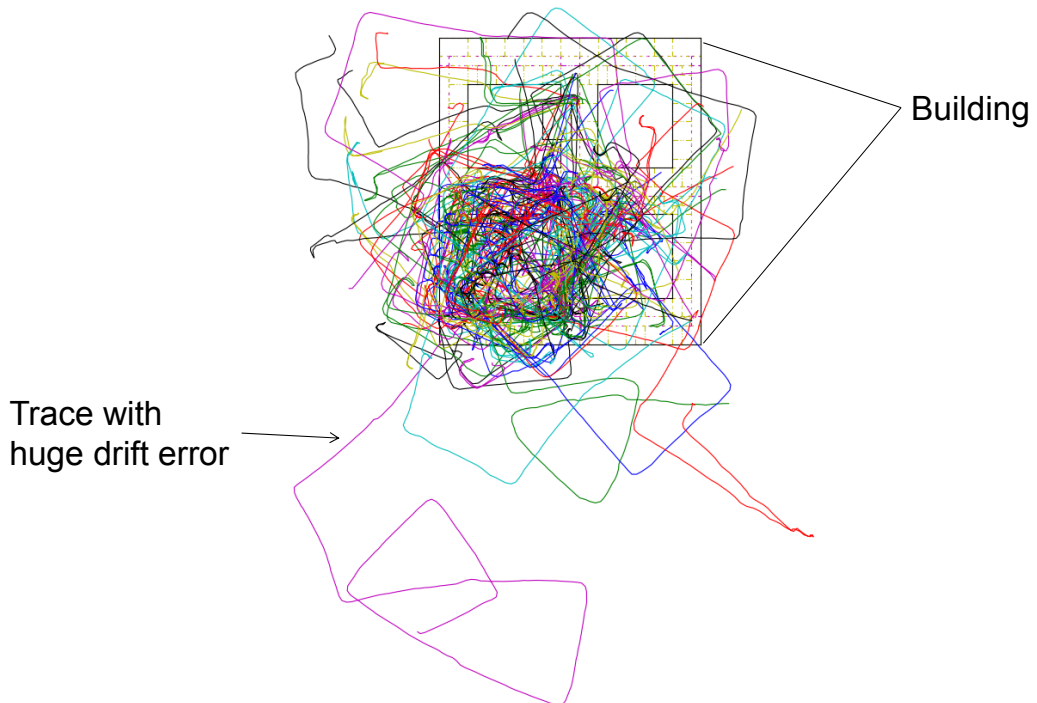


IMU:
Accelerometer &
Gyroscope

Output: Step sequence



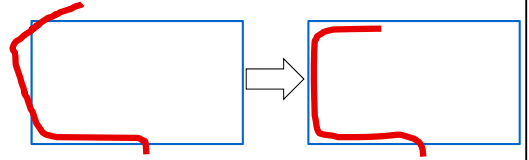
Example of Raw Traces



Trace-based Modelling

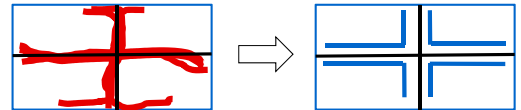
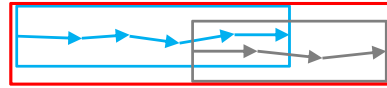
1. Trace correction

- Align segments along building axis



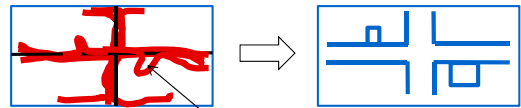
2. Hallway geometry estimation

- Bounding box grouping traces of one hallway into hallway areas
- Estimate hallway geometry with Gaussian filter from center line



3. Room geometry estimation

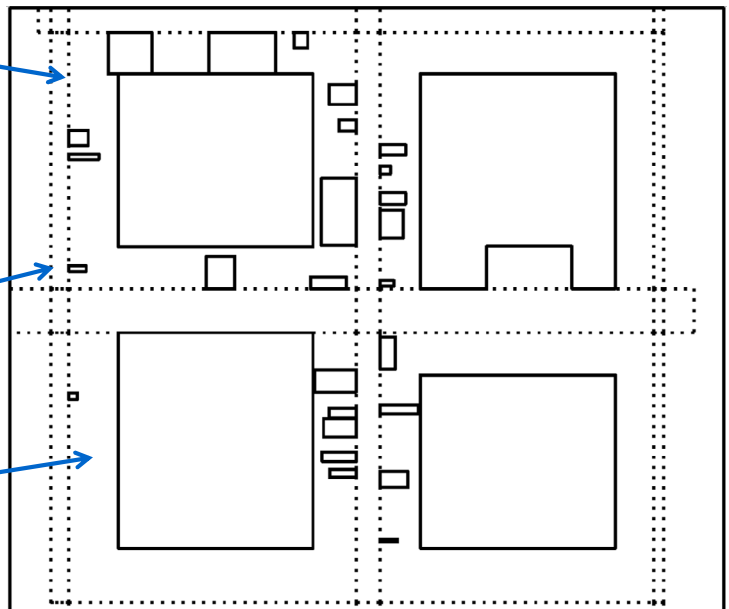
- Extract room segments
- Bounding box around room segments
- Match & merge bounding boxes of same room



Example of Trace-based Model



- Hallways mapped well →
- Room size often inaccurate →
- Too few rooms →



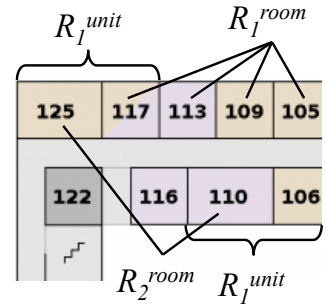
- Grammar of **regular language**

- **Single rooms incl. geometry**

- R_1^{room} ; 4.8 m length
- R_2^{room} ; 7.2 m length

- **Room units partially encoding topology**

- $R_1^{unit} = \{R_2 R_1\}$
- $R_2^{unit} = \{R_1 R_1\}$
- $R_3^{unit} = \{R_1 R_2 R_1\}$



- **Production rules** define how space can be filled by rooms:

- Space $\rightarrow R_1^{room}$ Space | R_2^{room} Space
- Space $\rightarrow R_1^{unit}$ Space | R_2^{unit} Space | R_3^{unit} Space
- Space $\rightarrow \epsilon$

- **Probabilities:** occurrence and adjacency of room types

Grammar-based Layout Generation

Basic Idea:

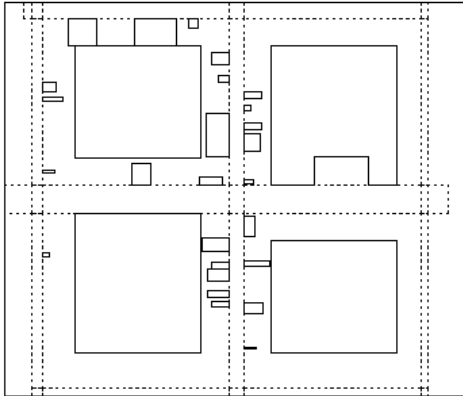
- Use grammar to generate all possible layouts
- Select layout that matches trace-based model best
- NP-hard problem!

- Use trace-based model to define **constraints**

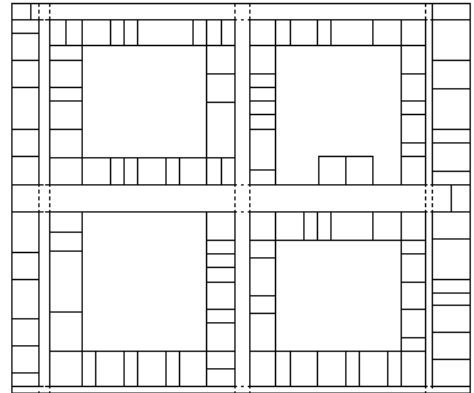
 - Don't place a wall inside an observed room
 - Place a wall in-between observed rooms

Heuristic:

- Use (**probabilistic**) grammar to generate **N most likely** layouts.
- Select layout that matches trace-based model best.



Trace-based model



Grammar-based model

250 traces (~22 km) in one building

- **Hallways detected properly** in both approaches
- Grammar-based model detects significantly **more rooms**
 - w/o grammar: 26 of 74
 - w/ grammar: 60 of 74

27

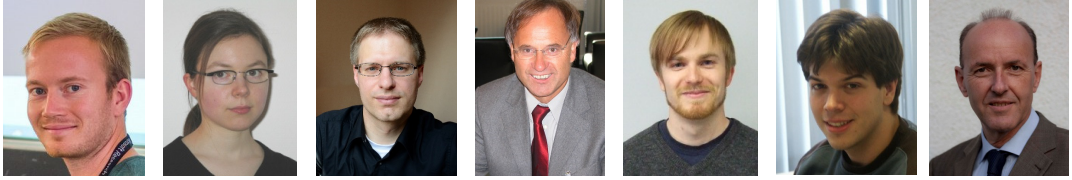
Summary

Public Sensing leverages mobile devices to gather sensor data

- Challenges:
 - Hide complexity, control quality of sensing
 - User acceptance: energy-efficiency & privacy
- Some Solutions
 - Virtual sensor abstraction
 - Model-driven sensing

Acknowledgements & Contact

Contributors



Supporters



Contact

kurt.rothermel@ipvs.uni-stuttgart.de

Links

<http://www.comnsense.de>

