SignalSLAM
Simultaneous Localization and Mapping with Mixed WiFi, Bluetooth, LTE and Magnetic Signals

Piotr Mirowski*, Tin Kam Ho, Saehoon Yi, Michael MacDonald
Statistics and Learning Research Department & Wireless Communications Research Department, Bell Labs, Alcatel-Lucent
Murray Hill, NJ, USA
piotr.mirowski@computer.org

Indoor Positioning and Indoor Navigation (IPIN) 2013
SignalSLAM: Outline

- **Goal:** RF signal fingerprinting
- **Scenario:** crowd-source RF mapping from the pocket
- **Methods**
  - Pedestrian dead reckoning
  - Calibration of PDR
  - GraphSLAM
  - Signal similarity
  - SignalSLAM
- **Experimental setup**
  - Android app
- **Results**
Goal: RF signal fingerprinting (WiFi, LTE, Bluetooth)

Access Point (WiFi, Bluetooth, 4G LTE small cell)

Sample Location

We want to build the RF signal map of the building, on a dense grid
**Goal:** RF Signal Fingerprinting (WiFi, LTE, Bluetooth)

Access Point (WiFi, Bluetooth, 4G LTE small cell)

Passively, from the pocket

Applications:

- WiFi / LTE localization
- Network deployment and optimization of 4G LTE (Long Term Evolution) small cells

We want to build the RF signal map of the building, on a dense grid
**SignalSLAM:** Outline

- **Goal:** RF signal fingerprinting
- **Scenario:** crowd-source RF mapping from the pocket
- **Methods**
  - Pedestrian dead reckoning
  - Calibration of PDR
  - GraphSLAM
  - Signal similarity
  - SignalSLAM
- **Experimental setup**
  - Android app
- **Results**
Scenario: Crowdsourced mapping from the pocket

Reconstruct Jane’s trajectory using only sensors embedded on a smartphone:

- GPS (when outdoor)
Scenario: Crowdsourced mapping from the pocket

Reconstruct Jane’s trajectory using only sensors embedded on a smartphone:

- GPS (when outdoor)
Scenario: Crowdsourced mapping from the pocket

Reconstruct Jane’s trajectory using only sensors embedded on a smartphone:

- GPS (when outdoor)
- inertial + magnetic: Pedestrian Dead Reckoning
Scenario: Crowdsourced mapping from the pocket

Reconstruct Jane’s trajectory using only sensors embedded on a smartphone:

- GPS (when outdoor)
- inertial + magnetic: Pedestrian Dead Reckoning
- NFC tags/QR code landmarks
Scenario: Crowdsourced mapping from the pocket

Reconstruct Jane’s trajectory using only sensors embedded on a smartphone:

- GPS (when outdoor)
- inertial + magnetic: Pedestrian Dead Reckoning
- NFC tags/QR code landmarks
- magnetic field perturbations
**Scenario:** Crowdsourced mapping from the pocket

Reconstruct Jane’s trajectory using **only sensors** embedded on a smartphone:

- GPS (when outdoor)
- inertial + magnetic: **Pedestrian Dead Reckoning**
- NFC tags/QR code landmarks
- magnetic field perturbations
- signal strength from:
  - WiFi routers
  - Bluetooth dongles
Scenario: Crowdsourced mapping from the pocket

Reconstruct Jane’s trajectory using only sensors embedded on a smartphone:

- GPS (when outdoor)
- inertial + magnetic: Pedestrian Dead Reckoning
- NFC tags/QR code landmarks
- magnetic field perturbations
- signal strength from:
  - WiFi routers
  - Bluetooth dongles
  - LTE 4G small cells
Scenario: Crowdsourced mapping from the pocket

Reconstruct Jane’s trajectory using only sensors embedded on a smartphone:

- GPS (when outdoor)
- inertial + magnetic: Pedestrian Dead Reckoning
- NFC tags/QR code landmarks
- magnetic field perturbations
- signal strength from:
  - WiFi routers
  - Bluetooth dongles
  - LTE 4G small cells
**Scenario:** Crowdsourced mapping from the pocket

Reconstruct Jane’s trajectory using only sensors embedded on a smartphone:

- GPS (when outdoor)
- inertial + magnetic: Pedestrian Dead Reckoning
- NFC tags/QR code landmarks
- magnetic field perturbations
- signal strength from:
  - WiFi routers
  - Bluetooth dongles
  - LTE 4G small cells
Scenario: Crowdsourced mapping from the pocket

Reconstruct Jane’s trajectory using only sensors embedded on a smartphone:

- GPS (when outdoor)
- inertial + magnetic: Pedestrian Dead Reckoning
- NFC tags/QR code landmarks
- magnetic field perturbations
- signal strength from:
  - WiFi routers
  - Bluetooth dongles
  - LTE 4G small cells

Close trajectory “loops” using GraphSLAM
**Scenario:** Crowdsourced mapping from the pocket

Reconstruct Jane’s trajectory using only sensors embedded on a smartphone:

- **GPS** (when outdoor)
- **inertial + magnetic:** Pedestrian Dead Reckoning
- **NFC tags/QR code landmarks**
- **magnetic field perturbations**
- **signal strength from:**
  - **WiFi routers**
  - **Bluetooth dongles**
  - **LTE 4G small cells**

**Close trajectory “loops” using GraphSLAM**

Exploit **signal similarity** using **SignalSLAM**
**Scenario:** Crowdsourced mapping from the pocket

Reconstruct Jane’s trajectory using only sensors embedded on a smartphone:

- GPS (when outdoor)
- inertial + magnetic: Pedestrian Dead Reckoning
- NFC tags/QR code landmarks
- magnetic field perturbations
- signal strength from:
  - WiFi routers
  - Bluetooth dongles
  - LTE 4G small cells

Close trajectory “loops” using **GraphSLAM**

Exploit signal similarity using **SignalSLAM**
Scenario: Crowdsourced mapping from the pocket

Reconstruct Jane’s trajectory using only sensors embedded on a smartphone:

- GPS (when outdoor)
- inertial + magnetic: Pedestrian Dead Reckoning
- NFC tags/QR code landmarks
- magnetic field perturbations
- signal strength from:
  - WiFi routers
  - Bluetooth dongles
  - LTE 4G small cells

Close trajectory “loops” using GraphSLAM

Exploit signal similarity using SignalSLAM
Scenario: Crowdsourced mapping from the pocket

Reconstruct Jane’s trajectory using only sensors embedded on a smartphone:

- GPS (when outdoor)
- inertial + magnetic: Pedestrian Dead Reckoning
- NFC tags/QR code landmarks
- magnetic field perturbations
- signal strength from:
  - WiFi routers
  - Bluetooth dongles
  - LTE 4G small cells

Close trajectory “loops” using GraphSLAM

Build a map of the RF signal

Exploit signal similarity using SignalSLAM
SignalSLAM: Outline

- Goal: RF signal fingerprinting
- Scenario: crowd-source RF mapping from the pocket
- Methods
  - Pedestrian dead reckoning
  - Calibration of PDR
  - GraphSLAM
  - Signal similarity
  - SignalSLAM
- Experimental setup
  - Android app
- Results
Methods: Pedestrian Dead Reckoning (PDR)

- **Real-time 3-axis orientation (Madgwick filter)**
  
Methods: Pedestrian Dead Reckoning (PDR)

- **Real-time 3-axis orientation (Madgwick filter)**
  

- **Step detection on vertical acceleration**
  
  [U. Steinhoff and B. Schiele (2010) “Dead reckoning from the pocket-an experimental study”, IEEE International Conference on Pervasive Computing and Communications (PerCom)]
**Methods: Pedestrian Dead Reckoning (PDR)**

- **Real-time 3-axis orientation (Madgwick filter)**
  

- **Step detection on vertical acceleration**
  
  [U. Steinhoff and B. Schiele (2010) “Dead reckoning from the pocket—an experimental study”, *IEEE International Conference on Pervasive Computing and Communications (PerCom)*]

- **Motion equation:**

  \[
  x_{1,t+1} = x_{1,t} + d_t \cos (\theta_t + \beta_t) \\
  x_{2,t+1} = x_{1,t} + d_t \sin (\theta_t + \beta_t) \\
  \theta_{t+1} = \theta_t + \phi_t
  \]

- **Calibrate:**
  
  step length \( d \), offset angle \( \beta \)
Methods: Least square calibration of PDR trajectories

Segment \([t_0, t_1]\) between two absolute position landmarks:

\[
\begin{align*}
\hat{x}_{1,t_0} &= x_{1,t_0} \\
\hat{x}_{2,t_0} &= x_{2,t_0} \\
\hat{x}_{1,t+1} &= \hat{x}_{1,t} + d \cos(\theta_t + \beta) \\
\hat{x}_{2,t+1} &= \hat{x}_{2,t} + d \sin(\theta_t + \beta)
\end{align*}
\]

Calibration of PDR: finding optimal step length \(d\), offset angle \(\beta\) (assuming that they are constant on that segment)
Methods: GraphSLAM

Observed constraint: \( Z_{t,t'} \)

Calculated displacement given current pose nodes:
\[
\hat{Z}_{t,t'} = \hat{Z}(x_t, x_{t'})
\]

Error for two nodes in graph:
\[
e_{t,t'} = Z_{t,t'} - \hat{Z}(x_t, x_{t'})
\]
Methods: GraphSLAM

Observed constraint: \( Z_{t,t'} \)

Calculated displacement given current pose nodes:
\[ \hat{Z}_{t,t'} = \hat{Z}(x_t, x_{t'}) \]

Error for two nodes in graph:
\[ e_{t,t'} = Z_{t,t'} - \hat{Z}(x_t, x_{t'}) \]

Objective function to minimize:
\[ F(x) = \sum_{(i,j) \in C} e_{ij}^T \Omega_{ij} e_{ij} \]
\[ x^* = \arg \min_x F(x) \]

- Linearize \( F \)
- Solve linear system
- Iterate until convergence on sequence \( x \)
Methods: Similarity in the signal-space

Distribution of RF signal $S$ from source $j$ around time $t$:

$$p_{t}^{(j)} = p(S | x_t)$$
Methods: Similarity in the signal-space

Distribution of RF signal $S$ from source $j$ around time $t$:

$$p_t^{(j)} = p(S|x_t)$$

Signal similarity between two segments at time $t$ and $t'$:

$$d_{t,t'} = \sum_j \left( KL(p_t^{(j)} | p_{t'}^{(j)}) + KL(p_{t'}^{(j)} | p_t^{(j)}) \right)$$

Kernel similarity between signals at time $t$ and $t'$:

$$k(x_t, x_{t'}) = e^{-\alpha d_{t,t'}}$$
**Methods: Similarity in the signal-space**

Distribution of RF signal \( S \) from source \( j \) around time \( t \):

\[
p_t^{(j)} = p(S|x_t)
\]

Signal similarity between two segments at time \( t \) and \( t' \):

\[
d_{t,t'} = \sum_j \left( KL(p_t^{(j)} | p_{t'}^{(j)}) + KL(p_{t'}^{(j)} | p_t^{(j)}) \right)
\]

Kernel similarity between signals at time \( t \) and \( t' \):

\[
k(x_t, x_{t'}) = e^{-\alpha d_{t,t'}}
\]

With multiple sources of signal:

\[
k_1(x_t, x_{t'}) \times k_2(x_t, x_{t'}) \times k_3(x_t, x_{t'})
\]

Source signals:
- \( WiFi \)
- 4G LTE from small cell
- Magnetic field strength
Methods: Kernel similarity in the signal-space

Kernel matrix in RF signal space using WiFi

- Time along trajectory of smartphone #1
- Time along trajectory of smartphone #2
- Time along trajectory of smartphone #3

Similar signal between phone #1 and phone #2 “hearing” the same WiFi

Self-similar signal at different points of time of trajectory due to loop closures (walking along same corridors)
Methods: Signal-similarity-based kernel regression

\[ \bar{x}_t = \frac{\sum_{t' \in N} x_{t'} k(S_t, S_{t'})}{\sum_{t' \in N} k(S_t, S_{t'})} \]

Predict new “signal-based” position of \( x_t \) using:

- positions of neighbours \( x_{t'} \),
- RF signal kernel similarity \( k(S_t, S_{t'}) \)
Methods: Signal-similarity-based kernel regression

\[ \overline{x}_t = \frac{\sum_{t' \in N} x_{t'} k(S_t, S_{t'})}{\sum_{t' \in N} k(S_t, S_{t'})} \]

Predict new “signal-based” position of \( x_t \) using:
- positions of neighbours \( x_{t'} \),
- RF signal kernel similarity \( k(S_t, S_{t'}) \)
Methods: Signal-similarity-based kernel regression

\[
\bar{x}_t = \frac{\sum_{t' \in N} x_{t'} k(S_t, S_{t'})}{\sum_{t' \in N} k(S_t, S_{t'})}
\]

Predict new “signal-based” position of \( x_t \) using:

- positions of neighbours \( x_{t'} \),
- RF signal kernel similarity \( k(S_t, S_{t'}) \)
Methods: Signal-similarity-based kernel regression

\[
\overline{x}_t = \frac{\sum_{t' \in N} x_{t'} k(S_t, S_{t'})}{\sum_{t' \in N} k(S_t, S_{t'})}
\]

Predict new "signal-based" position of \(x_t\) using:

- positions of neighbours \(x_t\),
- RF signal kernel similarity \(k(S_t, S_{t'})\)

"Signal edge" for GraphSLAM

Signal-based landmark for \(x_t\)
Methods: Signal SLAM

Modification of GraphSLAM:

• Iterate:
  • Signal kernel similarity calculation
  • Adding new “signal landmarks” and “signal edges”
  • Signal kernel regression of new node positions
  • GraphSLAM solution $O(N^2)$

• No need to represent complex uncertainty (distribution) at each point (e.g., particle filters)

• Simultaneous calculation of a global solution to entire pose sequence $x$

• Operates on any kind of RF signal that is stationary and on which similarity metric can be computed
SignalSLAM: Outline

- Goal: RF signal fingerprinting
- Scenario: crowd-source RF mapping from the pocket
- Methods
  - Pedestrian dead reckoning
  - Calibration of PDR
  - GraphSLAM
  - Signal similarity
  - SignalSLAM
- Experimental setup
  - Android app
- Results
Experiments: Android data acquisition app

Designed Android 2.3.3 app capable of recording:
- WiFi signal strength
- Bluetooth signal strength
- GPS fixes
- Google Maps WiFi-based geolocation and its accuracy
- Accelerometer, gyroscope and magnetometer readings
- Pedestrian dead reckoning
- NFC tag readings
- QR code tag readings

PDR recalibration, GraphSLAM, and SignalSLAM run offline (Matlab scripts).
Experiments: Low-cost landmarks using programmable tags

PDR positions can be manually reset using:

- QR codes can be printed on paper and generated using: http://qrcode.kaywa.com
- NFC tags can be generated using the NFC TagWriter app on Android

These tags are prepared and placed in advance throughout the building.

#refbl 40.684492,-74.401406
#refbl x45.1 y-10.5
**Experiments:** Low-cost landmarks using programmable tags

The NFC tags are located on multiple floors of the main building of Bell Labs (Murray Hill, NJ)
**SignalSLAM**: Outline

- **Goal**: RF signal fingerprinting
- **Scenario**: crowd-source RF mapping from the pocket
- **Methods**
  - Pedestrian dead reckoning
  - Calibration of PDR
  - GraphSLAM
  - Signal similarity
  - SignalSLAM
- **Experimental setup**
  - Android app
- **Results**
Results: Crowd-sourced SignalSLAM using multiple phones

PDR fails to reconstruct straight trajectories.

Strongly affected by magnetic field perturbations, due to low-quality IMU (compass and gyroscope).

PDR recalibration and GraphSLAM close some gaps.
Results: Crowd-sourced SignalSLAM using multiple phones

Single-phone reconstructions using a Samsung Galaxy Note 2

PDR better at reconstructing straight trajectories.

Less affected by magnetic field perturbations, due to higher-grade IMU (compass and gyroscope).

PDR recalibration and GraphSLAM close some gaps.

SignalSLAM joins trajectories along same corridor.
**Results:** Crowd-sourced SignalSLAM using multiple phones

Single-phone reconstructions using a Samsung Galaxy S4

- **PDR** better at reconstructing straight trajectories.
- Largest source of errors in PDR is placement of phone in the pocket.
- PDR recalibration and GraphSLAM close some gaps.
- SignalSLAM joins trajectories along same corridor.

---

Results: Crowd-sourced SignalSLAM using multiple phones

Multiple-phone reconstructions (joint SignalSLAM)

Exploit WiFi + Bluetooth signal similarity between trajectories of the 3 different phones
**Results:** Step-by-step comparison of SignalSLAM trajectories

Pairwise step-by-step distance between any two phones carried by the same experimenter: median <5m, <10m @ 90%
**Results:** Generation of dense RF signal maps

LTE RSRP (Verizon macro-cell)
RSRP (Reference Signal Received Power) recorded on a Samsung Galaxy Note 2 after SignalSLAM with landmarks, WiFi and Bluetooth

LTE RSRP (proprietary femto-cell)
RSRP (Reference Signal Received Power) recorded on a Samsung Galaxy S4 after SignalSLAM with landmarks, WiFi and Bluetooth
Thank you

- Contact address: piotr.mirowski@computer.org
- Read our full IPIN 2013 paper for further explanations: Piotr Mirowski, Tin Kam Ho, Saehoon Yi, Michael MacDonald “SignalSLAM: Simultaneous Localization and Mapping with Mixed WiFi, Bluetooth, LTE and Magnetic Signals”
**Goal:** Indoor RF mapping and localization (WiFi, LTE, ...)

In a training phase, we build the **RF signal map** of the building, by mapping RF fingerprints of the signal coming from all RF access points, on a dense grid.

During **tracking**, the WiFi signal received by the smartphone user is matched to the **RF signal map** (on the phone or on the server) and the user location is inferred using our **state-of-the-art localization** algorithm.
Automated fingerprinting: Example of LTE signal map

Bell Labs (Murray Hill), February 28, 2013

- Path 1 (fingerprinting)
- Path 2 (test run)
- Path 3 (test run)

LTE RSRP signal map

Choice of low-cost robotic platform

Robot trajectories
Results: PDR outdoors
Magnetic field perturbations indoor cause the **trajectory to drift over time**: classical problem in robotics (e.g., SLAM). This can be **corrected through optimization**, thanks to past/future trajectories, other users’ trajectories, landmarks (NFC, QR code and Bluetooth tags).
Localization: Example of prior results in a public space

Current accomplishments

- Complex public space: a busy shopping mall
  - Large space: 400m long, two levels, with atrium
  - Tracking (on Monday afternoon) done 3 days after fingerprinting (on Friday morning)
  - Accounting for sources of noise: people movements, antenna orientation, few RSSI samples, imprecise locations (5m error)
  - Unknown locations for all access points: (over 500 MAC addresses, some come and go)

- Potentials for improvement
  - Automated fingerprinting using a self-localizing autonomous robot
  - Semi-supervised fingerprinting of partially labelled trajectories of multiple users

Localization accuracy
- On the lower floor: median accuracy: 6.7m, accuracy @ 90%: 14.8m
- On the upper floor: median accuracy: 8m, accuracy @ 90%: 13.4m

Indoor Positioning and Indoor Navigation (IPIN) 2013

[Mirowski et al, IPIN 2011; Mirowski et al, JLBS 2012]
### Localization: Summary of performance

Results obtained using our KL-divergence kernel regression algorithm based on RSSI measurements

<table>
<thead>
<tr>
<th>Environment</th>
<th>Fingerprints with RSSI</th>
<th>Tracking accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#APs</td>
<td>#samples</td>
</tr>
<tr>
<td>Office (FT data)</td>
<td>4</td>
<td>130</td>
</tr>
<tr>
<td>Office (FT data)</td>
<td>22</td>
<td>130</td>
</tr>
<tr>
<td>Office corridor (BL)</td>
<td>4</td>
<td>120</td>
</tr>
<tr>
<td>Auditorium</td>
<td>6</td>
<td>120</td>
</tr>
<tr>
<td>Shopping mall (lower floor)</td>
<td>500</td>
<td>10</td>
</tr>
<tr>
<td>Shopping mall (upper floor)</td>
<td>500</td>
<td>10</td>
</tr>
<tr>
<td>Hamburg airport hall</td>
<td>17</td>
<td>40</td>
</tr>
<tr>
<td>Atrium (BL), robot automation</td>
<td>23</td>
<td>0.2m/s</td>
</tr>
</tbody>
</table>

Results obtained using our KL-divergence kernel regression algorithm based on AP visibility only

<table>
<thead>
<tr>
<th>Environment</th>
<th>Fingerprints, AP visibility</th>
<th>Tracking accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#APs</td>
<td>#samples</td>
</tr>
<tr>
<td>Office (FT data)</td>
<td>22</td>
<td>130</td>
</tr>
<tr>
<td>Office long corridor (BL)</td>
<td>130</td>
<td>1.4m/s</td>
</tr>
<tr>
<td>Shopping mall (lower floor)</td>
<td>500</td>
<td>10</td>
</tr>
<tr>
<td>Shopping mall (upper floor)</td>
<td>500</td>
<td>10</td>
</tr>
</tbody>
</table>

Indoor Positioning and Indoor Navigation (IPIN) 2013

[Mirowski et al, IPIN 2011; Mirowski et al, JLBS 2012]