The Jigsaw Continuous Sensing Engine for Mobile Phone Applications

Hong Lu, Jun Yang, Zhigang Liu, Nicholas D. Lane, Tanzeem Choudhury, Andrew T. Campbell

CS Department – Dartmouth College
Nokia Research Center – Palo Alto
Accelerometer

Light Sensor

Compass

GPS

Microphone

Camera

Gyroscope
Accelerometer
Bluetooth
Compass
CO₂
Barometer

GPS
Microphone
Camera
Gyroscope
EEG Sensors
However...
Pipeline Overview

Observe → Admission Control → Features → Classifier → Actions

Interpret
Main challenges of continuous sensing and inference on mobile phone:
- classification accuracy under various phone context
- resource efficiency
Robust Motion Based Physical Activity Classification

**Problems:** Calibration, Uncontrolled phone context

- specialized sensor
- fixed placement
Robust Motion Based Physical Activity Classification

Problems: Calibration, Uncontrolled phone context

In the real life …
One activity, Two distinct patterns

Accelerometer samples of cycling while phone (A) in the pant pocket, (B) in the backpack
Jigsaw Accelerometer Pipeline

Auto Calibration

Projection & Filtering

- Transition detection
- Projection
- 25 Features
- Classifier

Split-and-merge Classification

- Temporal smoothing

Features:
- Projection
- 25 Features
The calibration process
• provide parameters for normalization
  – Sensitivity (scaling factor), K
  – Offset, b
• hide hardware variance
• ensure the data quality
Jigsaw Accelerometer Pipeline

Auto Calibration

- Capture stationary moments
- Use linear least square estimator to calculate
  - Sensitivity
  - Offset

- 25 Features
- Classification
- Temporal smoothing
Jigsaw Accelerometer Pipeline

Auto Calibration

Calibration Results

- measurement error
  - without calibration 5% ~15%
  - user driven 0.55%
  - auto calibration
    - 0.76% N95
    - 0.58% iPhone 3G
Jigsaw Accelerometer Pipeline

- Estimate gravity direction by long term average
- Discard transition activity
- Project samples to global coordinate

Gravity Estimation & Projection

Transition detection

Projection

25 Features

Classifier

Temporal smoothing
Jigsaw Accelerometer Pipeline

- Split one activity into subclasses (different body positions).
- Merge the inference result back after inference.

Split-and-merge Classification

- Transition detection
- Projection
- 25 Features
- Classifier
- Temporal smoothing
Jigsaw Accelerometer Pipeline

25 Features

Transition detection

Projection

25 Features

Classifier

Temporal smoothing

Split-and-merge Classification
Jigsaw Accelerometer Pipeline

Auto Calibration

Projection & Filtering
- Transition detection
  - Projection
  - 25 Features
    - Classifier
    - Temporal smoothing

Split-and-merge Classification
Jigsaw Accelerometer Pipeline

Results:
• 16 people
• 5 body positions

<table>
<thead>
<tr>
<th>Action</th>
<th>DT</th>
<th>MG</th>
<th>SVM</th>
<th>NB</th>
</tr>
</thead>
<tbody>
<tr>
<td>stationary</td>
<td>95.19</td>
<td>98.07</td>
<td>97.68</td>
<td>96.19</td>
</tr>
<tr>
<td>walking</td>
<td>96.81</td>
<td>97.04</td>
<td>96.66</td>
<td>95.17</td>
</tr>
<tr>
<td>running</td>
<td>98.01</td>
<td>97.40</td>
<td>98.03</td>
<td>97.30</td>
</tr>
<tr>
<td>cycling</td>
<td>92.05</td>
<td>90.07</td>
<td>92.88</td>
<td>90.87</td>
</tr>
<tr>
<td>vehicle</td>
<td>90.52</td>
<td>87.47</td>
<td>90.29</td>
<td>89.83</td>
</tr>
<tr>
<td>Average</td>
<td>94.52</td>
<td>94.01</td>
<td>95.10</td>
<td>93.87</td>
</tr>
</tbody>
</table>
Jigsaw Accelerometer Pipeline

CPU Usage:

1 ~ 3 %  
0.9 ~ 3.7%
Light Weight Audio Classification

Main Problem: Computational Efficiency

• High data rate
• CPU usage on iPhone 3G
  ➢ Playing 256kbps AAC, ~8%
  ➢ Audio classification using Gaussian Mixture Model (GMM) classifier
     – 10 activities, ~20%
     – 20 activities, ~35%
Jigsaw Audio Pipeline

- Raw Waveform
- Sound Detection & Duty Cycling
- Feature Extraction
- Voice Detection
- Activity Sound Classification
- Smoothing Inference Result

Admission Control & Duty Cycling

Acoustic Features

Voice Classification

Voice

Similarity Detector

GMM Classifier

Temporal Smoothing
Jigsaw Audio Pipeline

Admission Control & Duty Cycling

Sound Detection & Duty Cycling

Acoustic Features

Voice Classification

Similarity Detector

GMM Classifier

Temporal Smoothing

Voice

Other

low duty cycling

continuous sensing

sound

silence

sound

silence
Jigsaw Audio Pipeline

Voice Detection

Admission Control & Duty Cycling

Acoustic Features

Voice Classification

Similarity Detector

GMM Classifier

Temporal Smoothing

Voice

Other

Temporal Smoothing

GMM Classifier

Similarity Detector

Voice Classification

Acoustic Features

Admission Control & Duty Cycling

Voice Detection
Jigsaw Audio Pipeline

Admission Control & Duty Cycling

Acoustic Features

Voice Classification

Similarity Detector

GMM Classifier

Temporal Smoothing

Voice Detection
Jigsaw Audio Pipeline

Activity Sound Classification
• 20-MFCCs feature
• GMM classifier

Admission Control & Duty Cycling

Acoustic Features

Voice Classification

Voice

Similarity Detector

Other

GMM Classifier

Temporal Smoothing
Jigsaw Audio Pipeline

- Admission Control & Duty Cycling
- Acoustic Features
- Voice Classification
  - Similarity Detector
  - GMM Classifier
- Temporal Smoothing
- Voice

\[ F = [f_1, f_2, f_3, \ldots, f_n] \]

label = “brushing teeth”
Jigsaw Audio Pipeline

Similarity Detector

Admission Control & Duty Cycling

Acoustic Features

Voice Classification

Voice

Similarity Detector

GMM Classifier

Temporal Smoothing

\[ F = [f_1, f_2, f_3 \ldots, f_n] \]

label = “brushing teeth”
Jigsaw Audio Pipeline

Admission Control & Duty Cycling

Acoustic Features

Voice Classification

Voice

Similarity Detector

Temporal Smoothing

F = \{f_1, f_2, f_3, \ldots, f_n\}

F' = \{f_1', f_2', f_3', \ldots, f_n'\}

cosine distance

\begin{align*}
\text{label} &= \text{“brushing teeth”}
\end{align*}
Jigsaw Audio Pipeline

Admission Control & Duty Cycling

Acoustic Features

Voice Classification

Voice

Similarity Detector

Temporal Smoothing

GMM Classifier

F = \[f_1, f_2, f_3 \ldots , f_n\]

F’ = \[f_1’, f_2’, f_3’ \ldots , f_n’\]

label = “washing hands”
Jigsaw Audio Pipeline

Admission Control & Duty Cycling

Acoustic Features

Voice Classification

Voice

Temporal Smoothing

Similarity Detector

GMM Classifier

F' = \[f_1', f_2', f_3' \ldots, f_n'\]

label = “washing hand”
Jigsaw Audio Pipeline

F' = [f1', f2', f3' ... , fn']

label = “washing hand”
Jigsaw Audio Pipeline

Admission Control & Duty Cycling

Acoustic Features

Voice Classification

Voice

GMM Classifier

Temporal Smoothing

Similarity Detector

F’ = [f1’, f2’, f3’ ... , fn’]
label = “washing hand”
Jigsaw Audio Pipeline

Admission Control & Duty Cycling

Acoustic Features

Voice Classification

Voice

Similarity Detector

GMM Classifier

Temporal Smoothing

\[ F' = [f_1', f_2', f_3' \ldots , f_n'] \]

label = “washing hand”
Jigsaw Audio Pipeline

Admission Control & Duty Cycling

Voice Classification

Similarity Detector

Voice

Temporal Smoothing

Similarity Detector

Acoustic Features

Other

GMM Classifier

CPU benchmark:
- similarity measure 0.02ms
- 7-class GMM classifier ~11ms
Jigsaw Audio Pipeline

Similarity Detector

Tuning the similarity threshold
- a tight threshold
  - invoke GMM more frequent
  - more accurate but costly
- a loose threshold
  - invoke GMM less often
  - more efficient by less accurate

\[
F' = [f_1', f_2', f_3' \ldots , f_n']
\]

threshold

\[
F = [f_1, f_2, f_3 \ldots , f_n]
\]

label = “washing hands”
Jigsaw Audio Pipeline

Admission Control & Duty Cycling → Acoustic Features → Voice Classification → Similarity Detector → Other → GMM Classifier → Temporal Smoothing

Similarity Detector

Accuracy vs. Percentage of GMM classification saved (%)

Accuracy:
- 1.0
- 0.9
- 0.8
- 0.7
- 0.6
- 0.5

Percentage of GMM classification saved (%):
- 0
- 20%
- 40%
- 60%
- 80%
- 100%
Jigsaw Audio Pipeline

Result

Classification Accuracy for different activity sounds
Jigsaw Audio Pipeline

CPU Usage:

7 ~ 17%  6 ~ 15%
Jigsaw Adaptive GPS Sensing

Main challenge: battery efficiency
Jigsaw Adaptive GPS Sensing

Main challenge: battery efficiency
Jigsaw Adaptive GPS Sensing

Hard to design a fixed duty cycle scheme that works for everyone

by Google Ridefinder
Jigsaw Adaptive GPS Sensing

The goal: fine tune the GPS sampling interval by:

1. Battery budget  
2. Tracking Duration  
3. User’s mobility pattern
Jigsaw Adaptive GPS Sensing

Markov Decision Process (MDP) based optimization.

Action = \{a \mid \text{sample every interval}(a), 1 \leq a \leq 6\},
for a = 1, 2 \ldots, 6, the interval(a) is 20min, 10min, 5min, 2min, 1min, 5s.
Jigsaw Adaptive GPS Sensing

Markov Decision Process (MDP) based optimization.

- Reward function,

\[
R \left( \left( e(i), t_i, m(i) \right) , a \right) =
\begin{cases} 
-Penalty_{nopower} & \text{if } e_{gps}(i) = 1, \ t_i < T \\
 k_m \cdot m(i) \cdot a & \text{otherwise}
\end{cases}
\]

- Policy Iteration algorithm is used to optimize the total reward over the time duration T.
The output of MDP is a policy table. For each (energy level, time tick, mobility level) tuple there is a corresponding sampling interval.
Jigsaw Adaptive GPS Sensing

Jigsaw Adaptive GPS Sensing in Action
Jigsaw Adaptive GPS Sensing

Evaluation

<table>
<thead>
<tr>
<th></th>
<th>stationary</th>
<th>walking</th>
<th>running</th>
<th>cycling</th>
<th>vehicle</th>
</tr>
</thead>
<tbody>
<tr>
<td>weekday</td>
<td>77.3%</td>
<td>10.5%</td>
<td>0.6%</td>
<td>0.4%</td>
<td>11.2%</td>
</tr>
<tr>
<td>weekend</td>
<td>38.1%</td>
<td>37.7%</td>
<td>1.8%</td>
<td>0.1%</td>
<td>22.3%</td>
</tr>
</tbody>
</table>

Mobility State Distribution in Location Traces
Jigsaw Adaptive GPS Sensing

Error vs. Power Tradeoff for Weekday Traces
Summary

- Jigsaw addressed the robustness and efficiency challenges of phone sensing and inference. It provides:
  - robust accelerometer activity classification
  - computational efficient audio classification
  - location tracking adaptive to battery budget, tracking duration, and user’s real-time mobility.

- Future Work
  - add more sensing modalities
  - high level inference combining multiple modalities
  - port to Google Android platform
Mobile Phone Sensing is the Next Big Thing!
Cheers!
Questions?