CS 528 Mobile and Ubiquitous Computing
Lecture 8a: Other Ubicomp Android APIs, Examples of Smartphone Sensing Apps

Emmanuel Agu
Announcement: Project 4

- Project 4 will be posted on Monday, October 30, 2017
- Will be due in 2 weeks from today (Thurs, November 9, 2017)
What other Android APIs may be useful for ubicomp?
Speaking to Android

https://developers.google.com/voice-actions/

- **Speech recognition:**
  - Accept inputs as speech (instead of typing) e.g. dragon dictate app?
  - Note: Requires internet access

- **Two forms**
  1. **Speech-to-text**
     - Convert user’s speech to text. E.g. display voicemails in text
  2. **Voice Actions**: Voice commands to smartphone (e.g. search for, order pizza)
Gestures

- **Gesture**: Hand-drawn shape on the screen
- **Example uses:**
  - Search your phone, contacts, etc by handwriting onto screen
  - Speed dial by handwriting first letters of contact’s name
  - Multi-touch, pinching
More MediaPlayer & RenderScript


- MediaRecorder is used to **record** audio
  - Manipulate raw audio from microphone/audio hardware, PCM buffers
    - E.g. if you want to do audio signal processing, speaker recognition, etc
    - **Example:** process user’s speech, detect emotion, nervousness?
  - Can playback recorded audio using MediaPlayer

- **RenderScript**
  - High level language for GPGPU
  - Use Phone’s Graphics Processing Unit (GPU) for computational tasks
  - Very few lines of code = run GPU code
  - Useful for heavy duty tasks. E.g. image, video processing
Wireless Communication

- Bluetooth
  - Discover nearby bluetooth devices
  - Communicating over bluetooth

- WiFi
  - Scan for WiFi hotspots
  - Monitor WiFi connectivity, Signal Strength (RSSI)
  - Do peer-to-peer (mobile device to mobile device) data transfers

Wireless Communication

- NFC:
  - Contactless technology
  - Transfer small amounts of data over short distances
  - **Applications**: Share spotify playlists, Google wallet
  - **Google wallet?**
    - Store debit, credit card on phone
    - Pay by tapping terminal
Telephony and SMS


- **Telephony:**
  - Initiate phone calls from within app
  - Access dialer, etc

- **SMS:**
  - Send/Receive SMS/MMS from app
  - Handle incoming SMS/MMS in app
Other 3<sup>rd</sup> Party Stuff

http://web.cs.wpi.edu/~emmanuel/courses/ubicomp_projects_links.html

- **MPAndroid**: Add charts to your app

![Chart showing data over months](image1.png)

- **Trepn**: Profile power usage and utilization of your app (CPU, GPU, WiFi, etc)
  - By Qualcomm

![Performance Graph](image2.png)
Other 3rd Party Stuff

http://web.cs.wpi.edu/~emmanuel/courses/ubicomp_projects_links.html

- **Programmable Web APIs:** 3rd party web content (e.g., RESTful APIs) you can pull into your app with few lines of code
  - **Weather:** Weather channel, yahoo weather
  - **Shared interests:** Pinterest
  - **Events:** Evently, Eventful, Events.com
  - **Photos:** flickr, Tumblr
  - **Videos:** Youtube
  - **Traffic info:** Mapquest traffic, Yahoo traffic

- **E.g. National Geographic:** picture of the day
AlcoGait
The Problem: Binge Drinking/Drunk Driving

- 40% of college students binge drink at least once a month
  - **Binge drinking defn:** 5 drinks for man, 4 drinks woman
- In 2013, over 28.7 million people admitted driving drunk
- Frequently, drunk driving conviction (DUI) results
Binge Drinking Consequences

- Every 2 mins, a person is injured in a drunk driving crash
- 47% of pedestrian deaths caused by drunk driving
- In all 50 states, after DUI -> vehicle interlock system
  - Also fines, fees, loss of license, lawyer fees, death
- Can we prevent DUI?

Vehicle Interlock system
Gait for Inferring Intoxication

- **Gait**: Way a person walks, impaired by alcohol
- Aside from breathalyzer, gait is most accurate bio-measure of intoxication
- The police also know gait is accurate
  - 68% police DUI tests based on e.g. walk and turn test
Can we test drinker’s before DUI? Prevent it?
  - At party while socializing, during walk to car

How? Alcogait smartphone app:
  - Samples accelerometer, gyroscope
  - Extracts accelerometer and gyroscope features
  - Classify features using Machine Learning
  - Notifies user if they are too drunk to drive
# Accelerometer Features Extracted

<table>
<thead>
<tr>
<th>Feature</th>
<th>Feature Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Steps</td>
<td>Number of steps taken</td>
</tr>
<tr>
<td>Cadence</td>
<td>Number of steps taken per minute</td>
</tr>
<tr>
<td>Skew</td>
<td>Lack of symmetry in one’s walking pattern</td>
</tr>
<tr>
<td><strong>Kurtosis</strong></td>
<td>Measure of how outlier-prone a distribution is</td>
</tr>
<tr>
<td>Average gait velocity</td>
<td>Average steps per second divided by average step length</td>
</tr>
<tr>
<td>Residual step length</td>
<td>Difference from the average in the length of each step</td>
</tr>
<tr>
<td>Ratio</td>
<td>Ratio of high and low frequencies</td>
</tr>
<tr>
<td>Residual step time</td>
<td>Difference in the time of each step</td>
</tr>
<tr>
<td>Bandpower</td>
<td>Average power in the input signal</td>
</tr>
<tr>
<td>Signal to noise ratio</td>
<td>Estimated level of noise within the data</td>
</tr>
<tr>
<td>Total harmonic distortion</td>
<td>“Determined from the fundamental frequency and the first five harmonics using a modified periodogram of the same length as the input signal” [22]</td>
</tr>
</tbody>
</table>

*Accelerometer gait features*
Posturography Sway Features

Posturography: clinical approach for assessing balance disorders from gait

Prior medical studies (Nieschalk et al) found that subjects swayed more after they ingested alcohol

Synthesized sway area features on 3 body planes and sway volume

Sway area computation: project values of gyroscope unto plane

E.g. XZ sway area:
- Project all observed gyroscope X and Z values in a segment an X-Z plane
- Area of smallest ellipse that contains all X and Z points in a segment is its XZ sway area
## Gyroscope Features Extracted

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Feature Description</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>XZ Sway Area</td>
<td>Area of projected gyroscope readings from Z (yaw) and X (pitch) axes</td>
<td>$XZ \text{ Sway Area} = \pi r^2$</td>
</tr>
<tr>
<td>YZ Sway Area</td>
<td>Area of projected gyroscope readings from Z (yaw) and Y (roll) axes</td>
<td>$YZ \text{ Sway Area} = \pi r^2$</td>
</tr>
<tr>
<td>XY Sway Area</td>
<td>Area of projected gyroscope readings from X (pitch) and Y (roll) axes</td>
<td>$XY \text{ Sway Area} = \pi r^2$</td>
</tr>
<tr>
<td>Sway Volume</td>
<td>Volume of projected gyroscope readings from all three axes (pitch, roll, yaw)</td>
<td>$Sway \text{ Volume} = \frac{4}{3} \pi r^3$</td>
</tr>
</tbody>
</table>
Steps for Training AlcoGait Classifier

Similar to Activity recognition steps we covered previously

1. Gather data samples + label them
   - 30+ users data at different intoxication levels
2. Import accelerometer and gyroscope samples into classification library (e.g. Weka, MATLAB)
3. Pre-processing (segmentation, smoothing, etc)
   - Also removed outliers (user may trip)
4. Extract features (gyroscope sway and accelerometer features)
5. Train classifier
6. Export classification model as JAR file
7. Import into Android app
Specific Issues: Gathering Data

- **Gathering alcohol data at WPI very very restricted**
  1. Must have EMS on standby
  2. Alcohol must be served by licensed bar tender
  3. IRB were concerned about law suits

- We improvised: used drunk buster Goggles

- “Drunk Busters” goggles distort vision to simulate effects of various intoxication (BAC) levels on gait

- Effects on goggle wearers:
  - Reduced alertness, delayed reaction time, confusion, visual distortion, alteration of depth and distance perception, reduced peripheral vision, double vision, and lack of muscle coordination.

- Previously used to educate individuals on effects of alcohol on one’s motor skills.
Different Sways? Swag?

- Different people sway different amounts even when sober
- Some people would be classified drunk even when sober (Swag?)
- Cannot use same absolute sway parameters for everyone
- Normalize!
  - Gather each person’s base data when sober
  - Divide possibly drunk gait features by sober features

\[
\frac{\text{drunk feature}}{\text{sober feature}}
\]

- Similar to how dragon dictate makes each reader read a passage initially
  - Learns unique inflexions, pronunciation, etc
- Classify absolute + normalized values of features
Box Plot of XZ Sway Area

- As subjects got more intoxicated, normalized sway area generally increased
AlcoGait Evolution

- Zach Arnold, Danielle LaRose
  - Initial AlcoGait prototype, accelerometer features (time, freq domain)
  - Real intoxicated gait data from 9 subjects, 57% accuracy
  - Best CS MQP 2015

- Christina Aiello
  - Data from 50 subjects wearing drunk busters goggles
  - Gyroscope features: sway area, 89% accurate
  - Best Masters grad poster 2016

- Muxi Qi (ECE)
  - Signal processing, compared 27 accelerometer features
AlcoWatch MQP: Using SmartWatch to Infer Alcohol levels from Gait

- **AlcoGait limitations:**
  - Users leave phones in drawers, bags, on table 50% of the time
  - Many women don’t have pockets, or carry their phones on their body

- **Alcowatch MQP: Detect alcohol consumption using smartwatch**
  - Classify accelerometer, gyroscope data

- **Students:** Ben Bianchi, Andrew McAfee, Jacob Watson
AlcoWear: Overview of How it Works

- Whenever user is walking, accelerometer + gyroscope data gathered simultaneously from smartphone + smartwatch
- Data sent to server for feature extraction classification
- Inferred BAC sent back to smartwatch, smartphone for display
AlcoWatch and AlcoGait Screens

AlcoWatch
(Smartwatch)

AlcoGait
(Smartphone)
AlcoWatch Features

- **AlcoGait Smartphone features**
  - Sway features (captures trunk sway)
  - Frequency-, Time-, Wavelet- and information-theoretic domain features

- **AlcoWatch Features**
  - Sway features
  - Arm velocity, rotation (pitch, yaw, roll) along X,Y,Z
Currently: NIH-Funded Study to Gather Intoxicated Gait Data from 250 Subjects

- Alcohol studies extremely tough at WPI (many rules)
  - **Rules:** Need EMS, bar tender, etc for controlled study
- Collaboration with physician, researchers at Brown university
- Gather intoxicated gait data from 250 subjects
- Controlled study:
  - Drink 1... walk
  - Drink 2... walk...
  - Etc
- Gather data, classify
BES Sleep App
Unobtrusive Sleep Monitoring

Unobtrusive Sleep Monitoring using Smartphones, Zhenyu Chen, Mu Lin, Fanglin Chen, Nicholas D. Lane, Giuseppe Cardone, Rui Wang, Tianxing Li, Yiqiang Chen, Tanzeem Choudhury, Andrew T. Campbell, in Proc Pervasive Health 2013

- Sleep impacts stress levels, blood pressure, diabetes, functioning

- Many medical treatments require patient records sleep
- Manually recording sleep/wake times is tedious
Unobtrusive Sleep Monitoring

- **Paper goal:** Automatically detect sleep (start, end times, duration) using smartphone, log it

- **Benefit:** No interaction, wear additional equipment,
  - Practical for large scale sleep monitoring

- Even a slightly wrong estimate is still very useful
Sleep Monitoring at Clinics

- Polysomnogram monitors (gold standard)
  - Patient spends night in clinic
- Lots of wires
- Monitors:
  - **Brain waves** using electroencephalography (EEG),
  - **Eye movements** using electrooculography,
  - **Muscle contractions** using electrocardiography,
  - **Blood oxygen levels** using pulse oximetry,
  - **Snoring** using a microphone, and
  - **Restlessness** using a camera
- Complex, impractical, expensive!
Commercial Wearable Sleep Devices

- Fewer wires
- Still intrusive, cumbersome
- Might forget to wear it

Can we monitor sleep with smartphone?
Insights: “Typical” sleep conditions

- Typically when people are sleeping
  - Room is Dark
  - Room is Quiet
  - Phone is stationary (e.g. on table)
  - Phone Screen is locked
  - Phone plugged in charging, off
Sense typical sleep conditions

- Use Android sensors to sense typical sleep conditions
  - **Dark:** light sensor
  - **Quiet:** microphone
  - **Phone is stationary (e.g. on table):** Accelerometer
  - **Screen locked:** Android system calls
  - **Phone plugged in charging, off:** Android system calls
Best Effort Sleep (BES) Model

- **BES model Features:**
  - Phone Usage features.
    - phone-lock (F2)
    - phone-off (F4)
    - phone charging (F3)
  - Light feature (F1).
  - Phone in darkness
  - Phone in a stationary state (F5)
  - Phone in a silent environment (F6)

- Each of these features are weak indicators of sleep
- Combine these into Best Effort Sleep (BES) Model
BES Sleep Model

- Assume sleep duration is a linear combination of 6 features

\[ Sl = \sum_{i=1}^{6} \alpha_i \cdot F_i, \alpha_i \geq 0 \]

- Gather data (sleep duration + 6 features) from 8 subjects
- Train BES model
- Formalize as a regression problem:

\[ \min_{\alpha_i} \sum_{j=1}^{4} (Sl^j - \sum_{i=1}^{6} \alpha_i \cdot F_i^j)^2 \]

Sleep duration, Weight for each feature, Feature (sum)
Regression?

- Gather sleep data (sleep duration, 6 features) from 8 subjects
- Fit data to line
  - **y axis** - sleep duration
  - **x-axes** – Weighted sum of 6 features
- **Weighted sum?** Determine weights for each feature that minimizes error
- Using line of best fit, in future sleep duration can be inferred from feature values
Results

<table>
<thead>
<tr>
<th>Feature</th>
<th>Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>Light ($F_1$)</td>
<td>0.0415</td>
</tr>
<tr>
<td>Phone-lock ($F_2$)</td>
<td>0.0512</td>
</tr>
<tr>
<td>Phone-off ($F_3$)</td>
<td>0.0000</td>
</tr>
<tr>
<td>Phone-charging ($F_4$)</td>
<td>0.0469</td>
</tr>
<tr>
<td>Stationary ($F_5$)</td>
<td>0.5445</td>
</tr>
<tr>
<td>Silence ($F_6$)</td>
<td>0.3484</td>
</tr>
</tbody>
</table>

TABLE 1: Weight coefficients for each feature in BES

Fig. 2: The reduction in sleep duration error for BES by incrementally adding stationary, silence, phone-lock, phone-charging, light and phone-off features, respectively.

Phone stationary (e.g. on table) most predictive .. Then silence, etc
Results

**Fig. 3:** Overall sleep duration error for BES compared to the three alternative sleep monitoring systems (SWP, Jawbone, Zeo).

**Fig. 5:** Comparison of estimated and actual sleep duration under BES for one representative study subject.
My actual Experience

- Worked with undergrad student to implement BES sleep model
- **Results:** About 20 minute error (+ or -) for 8-hour sleep
- Errors/thrown off by:
  - Loud environmental noise. E.g. garbage truck outside
  - Misc ambient light. E.g. Roommates playing video games