Whose Move is it Anyway?
Authenticating Smart Wearable Devices
Using Unique Head Movement Patterns

Sugang Li, Ashwin Ashok, Yanyong Zhang, Chenren Xu, Janne Lindqvist, Macro Gruteser

WINLAB

PerCom 2016
Project Highlights

- We design HeadBanger
- We achieve accurate user authentication
  - Experiment with 95 subjects
  - High TPR and Low FAR
  - Robust against Attack
- We build an running app on Google Glass
Personal Information Is in Your Wearables!

Location

Vital Sign

Permission Required Beyond This Point

Gaming Asset
Existing Approaches: Indirect Authentication

- Lack of independency
- Cumbersome
Existing Approaches: Built-in Authentication

- Limited Input Area
- Long Input Period
- Not Intuitive Pattern
Existing Approaches: Biometrics

Physical Biometrics:

- Additional Hardware
- Not always applicable for head-mounted device
Existing Approaches: Biometrics

Behavioral Biometrics:
Walking gait, arm swing, finger gesture, etc.

But, for head movements:
- Hard to collect long-term movement patterns
- Do not have high Degree of Freedom
Challenges

- Limited Input Area
- Invisible Input Pattern
- Constrained Recourses
- Low Degree of Freedom
- Lack of long-term Habit
Music-induced Head Movement
Music-induced Head Movement

- 30 Subjects
- Same movement and same music 30 times
Response Time Is Not Enough

Average Distances between the false users and the true user

Average distance between the true user and itself
Headbanger Rationales

Dist. Of the Same User
<<
Dist. Of Two Users
Headbanger Overview

Training Phase

Ave. Distance Computing

Ranking

Data Collection

Filtering

Distance Computing

Thresholding

Authentication

Reference Data

Template
Data Filtering

- Accelerometer Contains High Frequency Noise
- Head Movement is at Low Frequency ( < 5 Hz)
Headbanger Overview

Training Phase

Ave. Distance Computing

Ranking

Data Collection

Filtering

Distance Computing

Thresholding

Authentication

Template

Reference Data
**Dynamic Time Warping**

*Time-normalized distance* between A and B:

\[ D(A, B) = \left[ \frac{\sum_{s=1}^{k} d(p_s) \cdot w_s}{\sum_{s=1}^{k} w_s} \right] \]

- \[ d(p_s) \]: distance between \( i_s \) and \( j_s \)
- \( w_s > 0 \): weighting coefficient.

*Best alignment path* between A and B:

\[ P_1 = \arg \min_P (D(A, B)) \]

**Time Series A**

**Time Series B**
Headbanger Overview

Training Phase

Ave. Distance Computing

Ranking

Data Collection

Filtering

Distance Computing

Thresholding

Authentication

Template

Reference Data
Choose Representative Samples

- Compute the average distance to other samples
- Rank the samples based on their average distance
- Threshold can be expressed as:

\[ \text{Threshold} = \bar{d}_k + n \times \sigma_k \]
Repeatability & Similarity Experiment

Objectives:
- True user can login with high probability
- Different user do different movement
- Low computing cost

Setup:
- 30 subjects are involved
- Each of them design its own pattern
- Each of them performs it 40 times
Evaluation Metrics

- True Positive Rate
- False Accepted Rate
- Equal Error Rate

\[ \text{EER} = \text{TRR}(n) = \text{FAR}(n) \]
Impact of Distance Algorithm
Impact of Voting Scheme

![ROC curve showing impact of voting scheme with EER values.]
Impact of Training Size

![Graph showing the impact of training size on True Positive Rate and False Accepted Rate.](image-url)
Impact of Music Duration
Overall

![Bar chart showing EER (%) for different time durations: 4.43% for 10 Sec, 6.14% for 6 Sec, and 6.65% for 5 Sec.]
Let’s Attack it!

Subject A
On-beat
(a)

Subject B
Off-beat
(b)

Subject C
(c)
Let’s Attack It!

1. Watch the user’s movement video
2. Mimic the user’s movement
3. Headbanger displays the authentication result

One More Try!
## Attack Results

<table>
<thead>
<tr>
<th>Target</th>
<th># of Attackers</th>
<th># of Successful Attackers</th>
<th>Average # of Trials before 1(^{st}) Successful Attack</th>
<th>FAR (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>12</td>
<td>7</td>
<td>10</td>
<td>15.83</td>
</tr>
<tr>
<td>B</td>
<td>13</td>
<td>3</td>
<td>14</td>
<td>2.77</td>
</tr>
<tr>
<td>C</td>
<td>12</td>
<td>3</td>
<td>17</td>
<td>2.72</td>
</tr>
<tr>
<td>Overall</td>
<td>38</td>
<td>13</td>
<td>13</td>
<td>6.94</td>
</tr>
</tbody>
</table>
Prototyping

- Google Glass Development Kit
- Java Speech Tool Kit
- Fast DTW: $O(n^2) \rightarrow O(n)$
- Task pipelining

<table>
<thead>
<tr>
<th>Music Cue Duration (s)</th>
<th>Data processing latency (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>1.93</td>
</tr>
<tr>
<td>6</td>
<td>1.15</td>
</tr>
<tr>
<td>5</td>
<td>0.88</td>
</tr>
</tbody>
</table>
Conclusion

- We design Headbanger

- We Conduct Intensive Experiment
  - Repeatability, Robustness

- We develop a running App on Google Glass
Thank you! Questions?