

# 各種センサを統合した行動状態推定機能を持つライフログシステムの Android への実装と評価.

大久保 成晃 †

菊池 浩明 †

† 東海大学 情報通信学部 通信ネットワーク工学科

## 1 はじめに

携帯端末等に搭載されている GPS を用いて、現在位置に応じた情報提供をするロケーションサービスやライフログサービスが盛んになってきている。例えば Google latitude[1] では、利用者の居た場所を長期間記録できる。並木ら [2] は、Android 端末を用いて自転車の走行情報を利用した研究を行っている。しかしながら、ユーザの状態を考慮したよりきめ細かいサービスを提供するには、位置情報だけでは不十分である。例えば、歩行時ならば周辺店舗の情報は有益だが、電車内では無用である。

そこで、本研究では、Android 端末に掲載されている GPS、電子コンパス、加速度センサを用い、利用者の状態(停止、歩行、エレベータ、階段、電車)を推定して記録する Android アプリケーションを開発し、状態推定の精度を測る。

## 2 ライフログシステム “Navi”

Android 端末に搭載されるセンサの値を記録する専用アプリケーション “Navi” を開発する。“Navi” は、基本はログの出力であり、それに加え、センサの可視化、ログにコメント付随させる機能を持つ。ログは独自形式のテキストをバックグラウンドで出力するため、起動した後はアプリケーションを閉じて別の作業が可能である。図 1 は “Navi” の実行画面である。

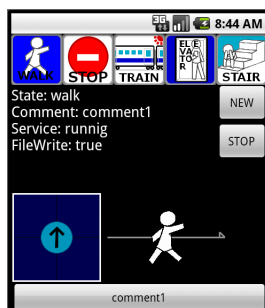


図 1: “Navi” 実行画面

Life-Logging System with status prediction based on series of sensors.

†Nariaki OHKUBO †Hiroaki KIKUCHI

†School of Information and Telecommunication Engineering, Tokai University

## 3 提案推定方式

### 3.1 予備実験

端末利用者の主な状態として、停止、歩行、エレベータ内、電車内、階段を考える。特徴量の性質を元に判定方式を定めるため、各状態のセンサ値を測定した。測定は 3 回ずつ、(1) 階段、高尾山、8 月 11 日、(2) 歩行、秋葉原、8 月 25 日、(3) エレベータ、秋葉原電気店、8 月 25 日に行った。Android 端末は腰ポケットに収納して測定した。測定に用いた端末は HTC 社製 Desire HD である。

### 3.2 測定結果と特徴

図 3,4,5 に各測定結果を示す。図 3 より、歩行時の Y 軸の加速度  $dy$  の変動が確認できる。左右が非対称なのは片側の腰ポケットに入れていたためである。

階段時は加速度の値の変化が歩行時と似ている。そのため、歩行時との明確な差が生じた Pitch の値に注目する。図 4 は、階段と歩行時の Pitch のヒストグラムである。

図 5 より、エレベータ内では、開始時と停止時に  $dy$  が一定の人工的な等加速度運動が確認できる。

電車内では利用者が動いていないので、Pitch と Roll の変動が少ない。他にはっきりとした特徴が見受けられず、わずかな加速度の動きを推定している。

### 3.3 提案方式

1. 歩行の判定。加速度の増減等に基づいた図 2 の判定アルゴリズムを用いる。予備実験により閾値 5.4 を決めて 1 歩を判定する。歩数が合計 4 歩を超えた際に歩行であると判定する。
2. 階段の判定。歩行と判定されたものから更に、Pitch  $p$  と Roll  $r$  の分散  $\sigma_p$  と  $\sigma_r$  の和が閾値 40000 を超えた際に階段であると判定する。
3. エレベータの判定。加速度を更に一度微分することにより、加速度の特徴を抽出し、微分間の時間差からエレベータであると判断する。

4. 電車の判定 . Pitch と Roll が動いておらず , かつ加速度が一定の増減をしていることから電車内であると判定する .
5. 停止 . 上記いずれの状態にも当てはまらない時と定める .

```

1 double[] y;//Y 軸加速度
2 boolean upWalk;//上昇中に真
3 double highY,lowY;//加速度の最高値,最低値
4 for(int i=0;i < y.length;i++){
5     if(upWalk){
6         if(highY <= y[i]){highY = y[i];}
7         else{lowY = y[i]; upWalk = false;}
8     }else{
9         if(lowY >= y[i]){lowY = y[i];}
10        else{
11            if(highY - lowY >= 5.4){walkCount++;}
12            highY = y[i];upWalk = true;
13        }
14    }
15 }

```

図 2: “歩行” 判定アルゴリズム

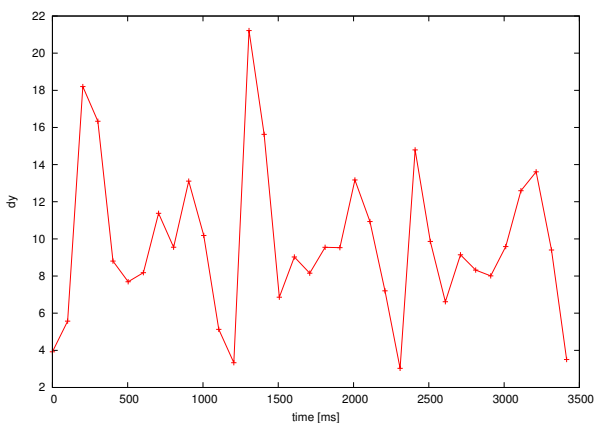


図 3: 歩行時の Y 軸加速度の変化

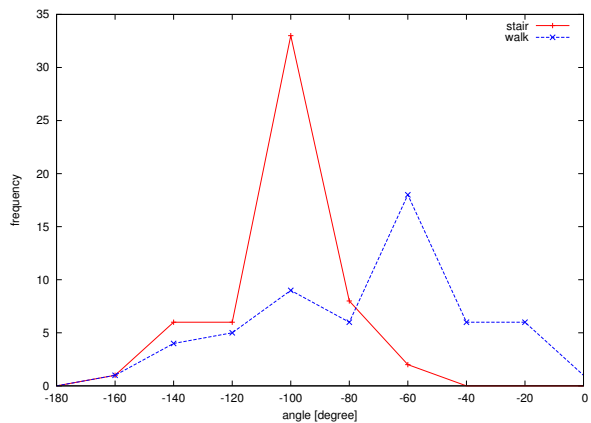


図 4: 階段と歩行時の Pitch ヒストグラム

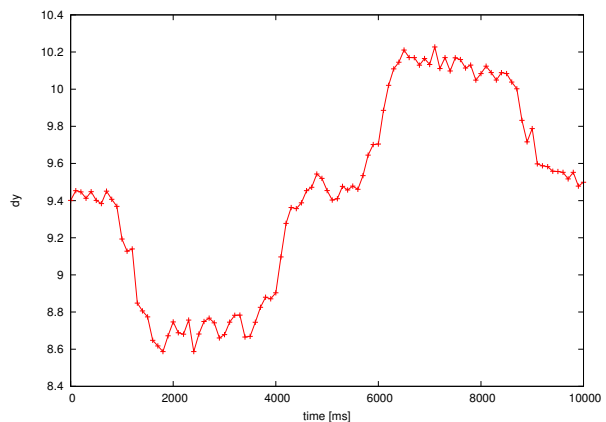


図 5: エレベータ内の Y 軸加速度の変化

判定結果より, 電車とエレベータの精度が極端に低いことが分かる . 電車内では慣性の法則により加速度の変化が少ないためである .

エレベータの判定は, 5 秒間は必要である . そのため, エレベータの判定方法は改善が必要である .

表 1: 提案方式の精度

真 \ 判定	walk	stair	train	elevator	stop	sum
walk	31	9	0	0	0	40
stair	0	40	0	0	0	40
train	0	0	22	14	4	40
elevator	0	0	2	36	2	40
stop	0	0	16	9	15	40
適合率 $p$	1.0	0.82	0.55	0.61	0.71	
平均 $\bar{p}$						0.72

### 3.4 判定精度

状態ごとに 5 秒間分の csv ファイルを 200 個用意し, 評価した結果を表 1 に示す .

## 4 おわりに

Android 端末のセンサの値から利用者状態を判定する方式を提案し, 実装に基づいて精度を評価した . 利用者のナビゲートを行うアプリケーションなどの応用を今後の課題とする .

### 参考文献

[1] Google latitude, (<https://www.google.com/latitude/b/0/> 2012-01-13 参照).

[2] 並木, 菅生, 戸辺, “自転車の走行情報利用に関する一検討”, 情報処理学会第 73 回全国大会, pp. 69-70, 2011.