

モバイル自己抑制のための階段コンテキスト認識 システムのパラメータ最適化

吉田 京平[†] 服部 峻[†]

[†] 室蘭工業大学大学院 〒 050-8585 北海道室蘭市水元町 27-1

E-mail: [†]13054053@mmm.muroran-it.ac.jp, hattori@csse.muroran-it.ac.jp

あらまし 近年、歩きスマホによる事故が多発しており、特に階段ではユーザが自己抑制しないと大きな事故に繋がる危険性が高い。階段における歩きスマホ事故を未然に防ぐため、コンテキスト・ウェアネス技術を用い、階段における歩きスマホを検知できるモバイルアプリケーションを提案する。加速度センサから得られた加速度値、及び、4つの閾値とウィンドウサイズをパラメータとして、「静止」、「歩行」、「階段昇り」、「階段降り」の4つのコンテキスト認識を行っている。また、各コンテキストの再現率・適合率が最大となるようなパラメータを求める実験を行った。

キーワード 自己抑制, 階段, 加速度センサ, コンテキスト・ウェアネス, モバイルアプリケーション

Parameter-Optimized Stairway Context-Awareness for Mobile Self-Controller

Kyohei YOSHIDA[†] and Shun HATTORI[†]

[†] Muroran Institute of Technology 27-1 Mizumoto-cho, Muroran, Hokkaido 050-8585, Japan

E-mail: [†]13054053@mmm.muroran-it.ac.jp, hattori@csse.muroran-it.ac.jp

Abstract Recently, texting while walking is frequently causing many accidents, and especially in the case of stairways, not exercising self-control by users has a high risk of big accidents. To prevent texting while walking on stairways, this paper proposes a mobile application that detects it based on context-awareness technologies. The proposed system uses acceleration value acquired from accelerometer with four threshold values and window size as parameters, and recognizes four kinds of contexts: standing, walking, upstairs, and downstairs. In addition, the experiment that optimizes the parameters to maximize the recall and precision of each context has been carried out.

Key words Self-Control, Stairway, Accelerometer, Context-Awareness, Mobile Application

1. はじめに

昨今のインターネットの発展により、様々な Web コンテンツが生まれた。例えば、YouTube 等の動画共有サービス、Amazon 等の EC (電子商取引)、Facebook 等の SNS が挙げられる。しかし、これらの Web コンテンツに夢中になるあまり、勉強が疎かになるなど、実生活への影響を懸念する声もある。そこで我々は、Web コンテンツの誘惑に負けてしまう人々の悩みを解決するために、従来のフィルタリング技術を応用して、自身のアクセス行為を制御 (自己抑制) できる固定端末向けのアプリケーションを作成した [1]。

しかし、近年はモバイル環境における利用が広がることで、周囲に人間がいる状況での利用など、人間が暮らす生活空間との関係が密になって来ている。そのため、従来のような人間と端末だけの関係性では見られなかった、新たな問題が発生して

いる。例として、近年事故が増加傾向にあり、社会問題となりつつある「歩きスマホ問題」が挙げられるだろう。現状、歩きスマホ問題における事故は、駅のホームから転落する事故が多いが、柵を設置することで物理的な対策を行うことができる。しかし、階段ではユーザ自身で歩きスマホを自己抑制しないと転倒事故を起こす危険性が高く、巻き込み事故など周囲に影響を与える大きな事故も起こし易い。

以上のことから、歩きスマホ問題のような、従来の固定端末では見られない新しい問題に対して、モバイル環境向けの新しい自己抑制が必要であると考えられる。そこで、階段における歩きスマホ事故を未然に防ぐため、端末が階段におけるユーザの状況 (階段コンテキスト) を認識するコンテキスト・ウェアネスの技術を用い、ユーザが歩きスマホをしながら階段を移動したことを検知し、画面にアラートを発することで自粛を促すモバイルアプリケーションを提案する (図 1)。



図 1 階段における歩きスマホの自粛を促す様子 (イメージ)

Fig. 1 Alerting a user for texting while walking on stairways.

2. 提案手法

2.1 システム概要

本稿における階段コンテキスト認識システムは、図 2 に示す処理を繰り返すことで、ユーザの移動コンテキスト (空間×アクション) を認識する。なお、認識を行う移動コンテキストは、any (どこでも) × 静止、平坦な道×歩行、階段×昇り、階段×降りの 4 種類である。また、階段での昇り降りを区別するのは、例えば降りは昇りと比べて高所からの転落による大怪我を起こし易いため、降りだけ危険に感じるなど、ユーザによって自己抑制したい状況が異なるからである。

まず処理 1 では、加速度センサが内蔵されている Android 端末を持って静止、歩行、昇り、降りなどのアクションを行う。次に処理 2 で、各アクションによって端末上の座標系における加速度値が取得され、これを世界座標系に変換して鉛直方向の加速度値を取得する。処理 3 では、加速度センサから得られた鉛直方向の加速度値をコンテキスト推定機構に送ると、コンテキスト推定が実施される。最後に処理 4 で推定結果を画面に出し、処理 1 に戻る。

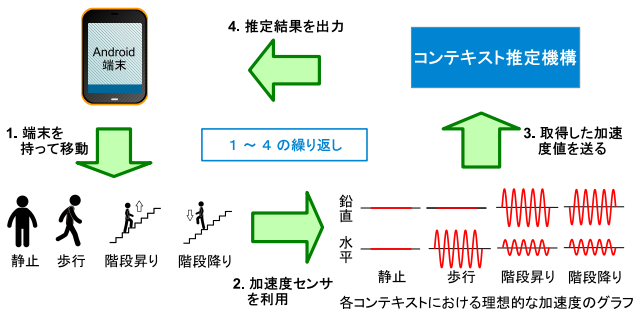


図 2 システムの全体像

Fig. 2 An overview of the proposed system.

2.2 各コンテキストの認識手法

本節では、図 2 におけるコンテキスト推定機構の詳細について説明する。コンテキスト推定機構では、鉛直方向の加速度値とウィンドウを利用して、各コンテキストの認識を行っている。ウィンドウは加速度値を格納する配列であり、ウィンドウサイズ (以下、WS) は 30 点である。また、サンプリング間隔はいつでも 60ms 間隔以内ではあるが不定である。図 3 に、加速度値とウィンドウを利用した各コンテキストの認識手法を示す。

各コンテキストには閾値が設定されており、これをウィンドウに含まれる加速度値の最大値・最小値と比較し、コンテキスト認識を行う。まず静止の場合、閾値は最大値が 0.5 以下かつ最小値が -0.5 以上のときである。ウィンドウ内の最大値と最小値はそれぞれ 0.4 と 0.0 であり、閾値を超えないため、静止と認識される。歩行は、最大値の閾値が 0.5 以上かつ 2.0 以下を満たし、さらに最小値の閾値が -2.0 以上かつ -0.5 以下を満たすときである。ウィンドウ内の最大値は 0.6 で、最小値は -0.8 であるので、歩行と認識される。昇りの閾値は、最大値が 2.0 以上を満たすときである。ウィンドウ内の最大値は 2.2 で、最小値は 0.0 であるが、階段昇りの認識には最小値は無関係であるため、最大値が 2.0 を超えてさえいれば階段昇りと認識される。一方、降りの閾値は、最小値が -2.0 以下を満たすときである。ウィンドウ内の最大値は 1.6 で、最小値は -3.1 であるが、階段降りの認識には最大値は無関係であるため、最小値が -2.0 を下回っていれば階段降りと認識される。

しかし、実際の階段は複数段あるため、図 3 のような 1 段分の波形にはならず、振幅の大きな波が連続して観測される。昇りと降りは共に加速度値の振幅が大きいため、例え実際のコンテキストが昇りであったとしても、ウィンドウ内の最大値が 2.0 を超えた直後に最小値が -2.0 を下回り、再び最大値が 2.0 を超える状況や、最大値・最小値が同時に昇り降りの閾値を満たす状況が発生する。そこで状態遷移を導入し、昇りと降りは互いに行き来しないようにすることで、昇りと降りの認識を交互に繰り返すのを防いでいる。つまり、始めに昇りと認識した場合、その後も昇りと認識され続けるようになっている。これは同時に、始めの認識を間違えた場合、その後の認識も間違い続けることを意味している。また、階段で急に昇りから降りへ方向転換した場合などにも、その変化を正確に認識するのは出来なくなる。図 4 に状態遷移図を示す。

なお、本システムでは各コンテキストの閾値を 4 つに分類している。歩行と静止を区別する最大値側の閾値を max-in、同じく最小値側の閾値を min-in とし、歩行と昇りの閾値を max-out、歩行と降りの閾値を min-out とする。max-in と min-in は歩行と静止の認識に影響を与えるが、昇りと降りの認識には無関係である。一方、max-out は昇りと歩行に影響を与えるが、静止と降りには影響を与えず、min-out は降りと歩行に影響を与えるが、静止と昇りには影響を与えない。

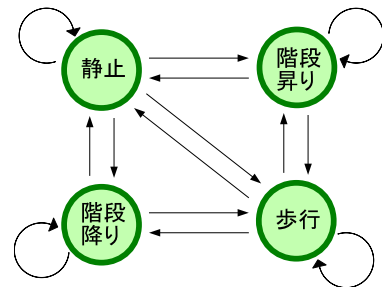


図 4 各コンテキストの状態遷移図

Fig. 4 A state transition diagram of four kinds of contexts.

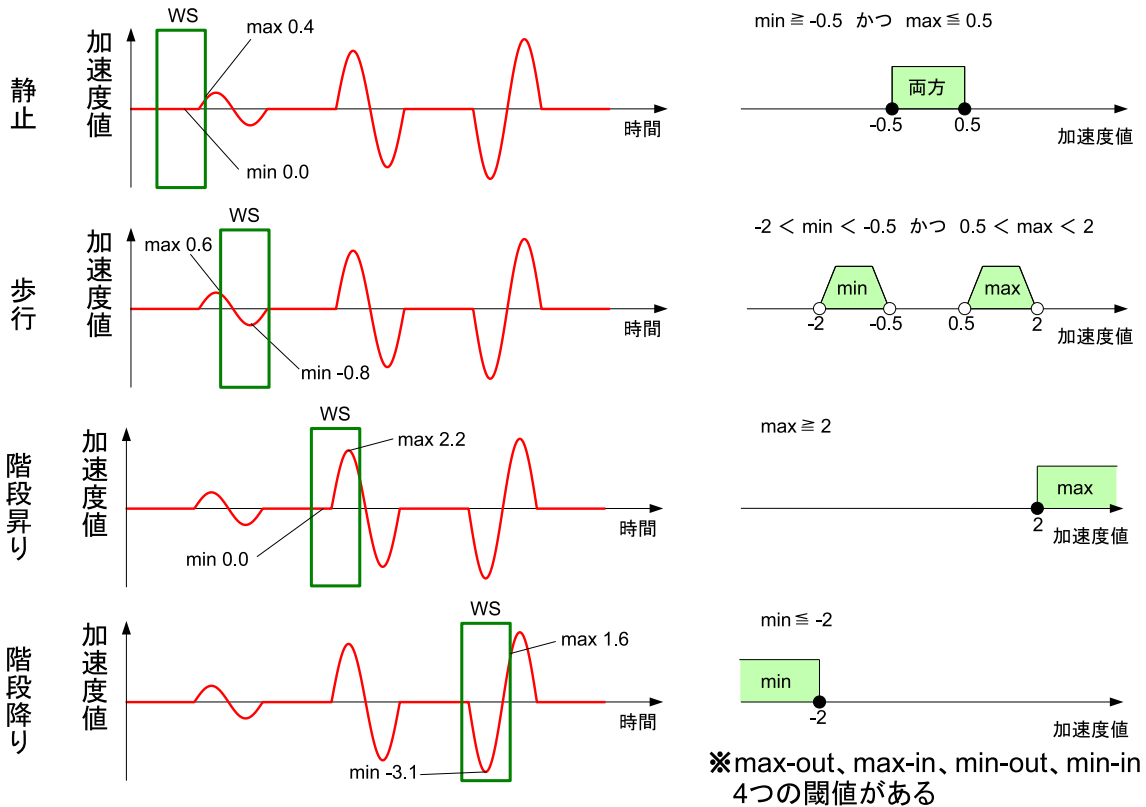


図 3 各コンテキストの認識手法と閾値について

Fig. 3 The proposed method to recognize each context with four threshold values and window size.

3. 評価実験

本システムを実装した端末を持ち歩く者と、実際にコンテキストが切り替わるタイミングを計測する者の2名で行い、大学校舎1階から4階を10往復して、階段昇り、降りのデータを10本ずつ得た。なお、役割の交代を行わなかったため、今回取得したデータはいずれも同じ人物(第1著者)のものである。表1に各コンテキストに対する認識割合を示す。但し、階段の踊り場は歩行とする。実験時は、加速度センサが内蔵されているAndroid端末(本実験ではASUS社のEee Pad TF201を使用)を持ち歩き、データの取得を行った。計測場所における階段の形状は、直線の階段→踊り場(階段の連結部分)→直線の階段を繰り返す形状となっている。本研究では、踊り場のアクションを歩行として定義し、昇り、降りのデータを10本ずつ取得した。昇りのデータ取得ルートは、1階廊下(静止)→1階廊下(歩行)→1階踊り場(歩行)→階段(昇り)→1.5階踊り場(歩行)→階段(昇り)→2階踊り場(歩行)→...→3.5階踊り場(歩行)→階段(昇り)→4階踊り場(歩行)→4階廊下(歩行)→4階廊下(静止)となっており、途中で立ち止まることはない。降りのルートは昇りと逆で、4階廊下(静止)が起点となり、1階廊下(静止)が終点となる。また、階段はいずれも階段(降り)となる。取得するデータは、鉛直方向、水平方向の加速度値であり、テキストファイルとして出力される。このテキストファイルをPC上に実装した本システムに流すと、5つのパラメータ(WS, max-in, min-in, max-out, min-out)

に応じてコンテキスト認識を行い、認識割合を出力する。以降に示す実験結果は、いずれもここで出力されたデータを用いる。まず、本システムの初期設定での(WS=30, max-in=0.5, min-in=-0.5, max-out=2.0, min-out=-2.0)認識割合を表1, 表2に示す。

表 1 初期設定における再現率

Table 1 Recall ratios in the default setting.

システム \ 実際	(any) 静止	(平坦) 歩行	(階段) 昇り	(階段) 降り
(any) 静止	71.28%	26.02%	1.73%	0.47%
(平坦) 歩行	0.01%	12.52%	67.93%	19.54%
(階段) 昇り	0.00%	0.00%	98.28%	1.72%
(階段) 降り	0.00%	1.00%	59.22%	39.78%

表 2 初期設定における適合率

Table 2 Precision ratios in the default setting.

システム \ 実際	(any) 静止	(平坦) 歩行	(階段) 昇り	(階段) 降り
(any) 静止	99.92%	0.08%	0.00%	0.00%
(平坦) 歩行	33.82%	64.15%	0.00%	2.03%
(階段) 昇り	0.18%	32.05%	37.78%	29.99%
(階段) 降り	7.90%	52.40%	4.31%	35.38%

再現率 (Recall) と適合率 (Precision) はそれぞれ以下の式で計算される。

$$\text{再現率} := \frac{\text{システムが認識した正解合計時間}}{\text{実際のコンテキストの合計時間}}$$

$$\text{適合率} := \frac{\text{システムが認識した正解合計時間}}{\text{システムが認識した合計時間}}$$

3.1 パラメータ最適化

続いて、再現率、適合率が最大となるようなパラメータを求める実験を行った。しかし、表 1、表 2 にあるような複数の数値では比較が難しいことから、1つの数値に集約を行って比較を容易にするための処理を行った。その際、表の太字部分を使って重み付け（静止×0.2+歩行×0.2+昇り×0.3+降り×0.3）を行うことで、再現率、適合率を一つの数値にそれぞれまとめ、さらにそれらを統合するために F 値を利用した。この F 値が最大となるようなパラメータを最適化されている状態とする。パラメータの可動範囲は、

- $20 \leq \text{WS} \leq 50$ (5 刻み)
- $0.1 \leq \text{max-in} \leq 1.0$ (0.1 刻み)
- $-1.0 \leq \text{min-in} \leq -0.1$ (0.1 刻み)
- $1.5 \leq \text{max-out} \leq 4.0$ (0.1 刻み)
- $-4.0 \leq \text{min-out} \leq -1.5$ (0.1 刻み)

とする。パラメータを変動させた結果、F 値が最大となったパラメータの組み合わせは、WS=40, max-in=1.0, min-in=-1.0, max-out=3.9, min-out=-2.1 であった。このときの再現率を表 3 に、適合率を表 4 に示す。

表 3 パラメータ最適化後の再現率

Table 3 Recall ratios after optimizing parameters.

システム \ 実際	(any) 静止	(平坦) 歩行	(階段) 昇り	(階段) 降り
(any) 静止	88.83%	9.48%	0.00%	1.69%
(平坦) 歩行	0.98%	37.49%	15.82%	45.71%
(階段) 昇り	0.00%	0.24%	70.49%	29.28%
(階段) 降り	0.00%	3.01%	0.00%	96.99%

表 4 パラメータ最適化後の適合率

Table 4 Precision ratios after optimizing parameters.

システム \ 実際	(any) 静止	(平坦) 歩行	(階段) 昇り	(階段) 降り
(any) 静止	95.64%	4.36%	0.00%	0.00%
(平坦) 歩行	5.18%	91.61%	0.47%	2.74%
(階段) 昇り	0.00%	23.93%	76.07%	0.00%
(階段) 降り	0.21%	36.48%	29.34%	33.98%

3.2 ウィンドウサイズ (WS) 以外のパラメータ変動

各々のパラメータが再現率・適合率に及ぼす影響を検証するため、パラメータ最適化後のパラメータを使って、一部のパラメータを変動させた。ここでは、階段に影響を及ぼすパラメータである max-out と min-out のみを変動させた様子を図 5～10 にそれぞれ示す。なお図中の×印は、パラメータ最適化後に際するパラメータの位置を表している。また、max-in, min-in

のパラメータは、静止と歩行の認識には関与したが、昇りと降りには影響を及ぼさなかったため、本稿では割愛する。

まず階段に関しては、図 5 から、昇りの再現率が高くなるのは、max-out, min-out の両方が小さいときであるとわかる。これは、昇りに関する閾値の条件が緩くなり、かつ降りに関する閾値の条件が厳しくなることで、全体的に昇りと認識し易くなったものと考えられる。本システムは加速度値の振幅が正に大きく振れ、ウィンドウ内の最大値が閾値の max-out を超えることで昇りと認識するため、max-out が小さいほど最大値が超え易くなり、階段昇りと認識をし易くなる。もし、max-out の閾値が高すぎる場合は、ウィンドウ内の最大値が max-out を超えきれないため、昇りと認識され難くなり、再現率が低下する。また、min-out の閾値が大きい場合、max-out の閾値を超えきれずその後の加速度値が負に大きく振れて、ウィンドウ内の最小値が min-out を下回り易くなり、降りと誤認識する可能性が高くなる。min-out の閾値を小さくし、降りと認識しづらくすることで、その後、再び加速度値が正に振れたときに昇りと認識させ易くなり、結果昇りの再現率が向上する。上記は、降りの再現率でも同様のことが考えられる。図 7 から、降りの再現率が高くなるのは、max-out, min-out の両方が大きいときであり、昇りの再現率とは真逆となっている。これは min-out が大きいほど、降りの閾値である min-out を下回り易くなるので、降りの再現率が上昇するものと考えられる。また、max-out が大きいほど、昇りと認識しづらくなるので、昇りの誤認識が減ることから、降りの再現率が上昇すると考えられる。一方、適合率の方は、図 5 と図 6、図 7 と図 8 をそれぞれ比較することで、再現率と適合率がトレードオフの関係になっていることがわかる。これは、例えば昇りの場合、閾値を緩くして昇りと認識させ易くすると、取りこぼしが少なくなり再現率が高くなるが、昇り以外のコンテキストも昇りと認識し易くなるため、適合率が下がってしまうものと考えられる。

次に、歩行の再現率に関しては、図 9 から、max-out が大きく、min-in が小さい、つまり絶対値が大きくなるほど再現率が高くなるということがわかる。これは、歩行と認識する条件が緩くなるからと考えられる。昇りと降りに関する閾値の条件が厳しくなることで、加速度値がある程度大きい振幅となっても、昇り、降りと認識され難くなり、結果歩行と認識し易くなる。一方、適合率の方は、図 10 から、再現率が最も高くなるパラメータのとき、再現率が低くなっていることがわかる。また、max-out, min-out の絶対値が 2.5 付近のとき、適合率が高くなる傾向であるとわかる。これは、絶対値が小さすぎると昇りや降りと認識し易くなるため、歩行の認識がされ難くなり、逆に絶対値が大きすぎると歩行の認識が増えすぎるため、適合率が下がると考えられる。しかし、適合率が最大となるのは、max-out が 4.0 付近のときであるため、昇りと認識し難くすることで、歩行の適合率を高められることがわかる。

以上をまとめると、max-out と min-out を両方大きくすると昇りの適合率と降りの再現率が向上し、両方小さくすると昇りの再現率と降りの適合率が向上する。これらは、それぞれトレードオフの関係であるため、パラメータの変化だけでは昇り、

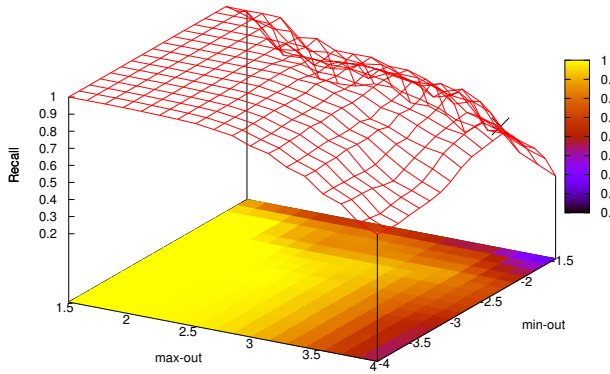


図 5 WS=40,max-in=1.0,min-in=1.0 における昇りの再現率
Fig. 5 Recall of upstairs with WS=40,max-in=1.0,min-in=1.0.

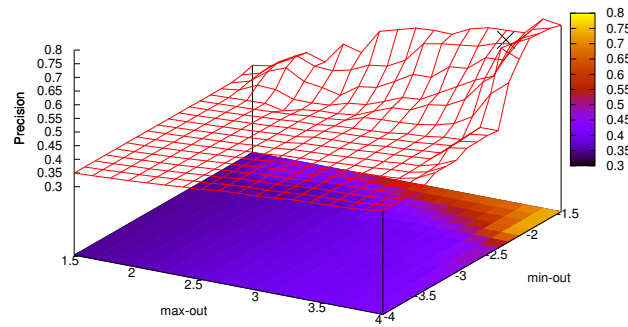


図 6 WS=40,max-in=1.0,min-in=1.0 における昇りの適合率
Fig. 6 Precision of upstairs with WS=40,max-in=1.0,min-in=1.0.

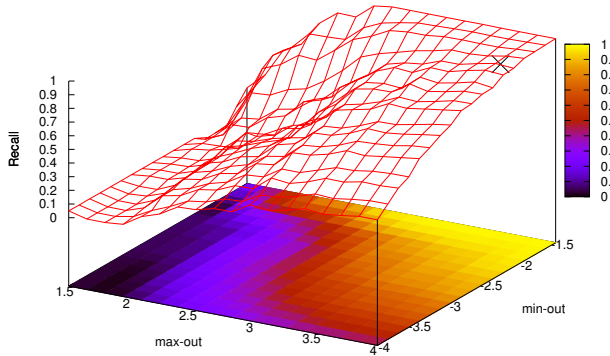


図 7 WS=40,max-in=1.0,min-in=1.0 における降りの再現率
Fig. 7 Recall of downstairs with WS=40,max-in=1.0,min-in=1.0.

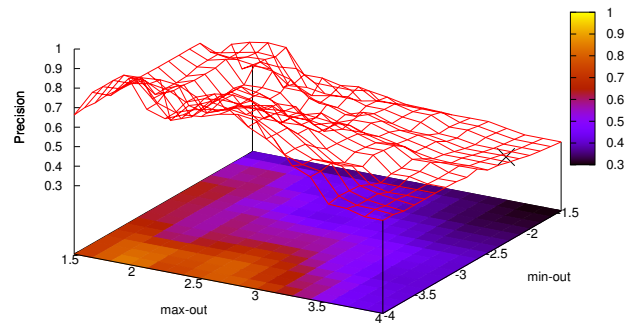


図 8 WS=40,max-in=1.0,min-in=1.0 における降りの適合率
Fig. 8 Precision of downstairs with WS=40,max-in=1.0,min-in=1.0.

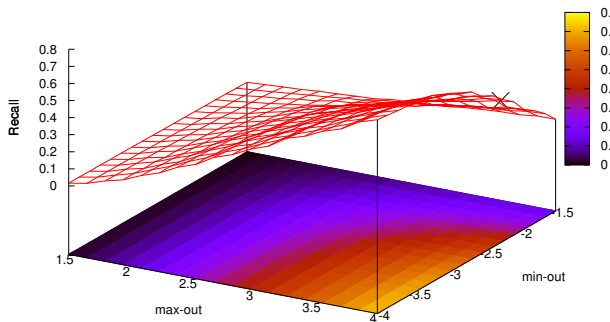


図 9 WS=40,max-in=1.0,min-in=1.0 における歩行の再現率
Fig. 9 Recall of walking with WS=40,max-in=1.0,min-in=1.0.

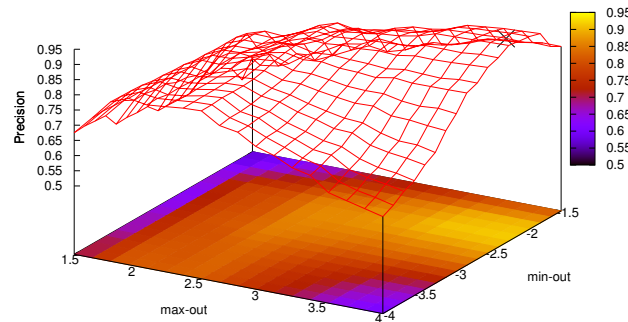


図 10 WS=40,max-in=1.0,min-in=1.0 における歩行の適合率
Fig. 10 Precision of walking with WS=40,max-in=1.0,min-in=1.0.

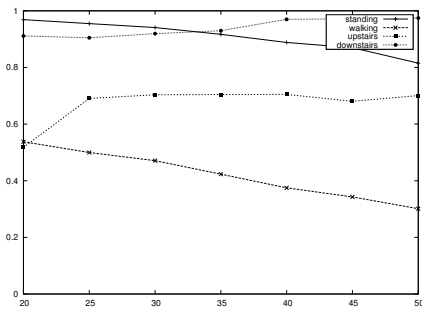


図 11 WSのみ変動させたときの再現率
Fig. 11 Recall in varying only WS.

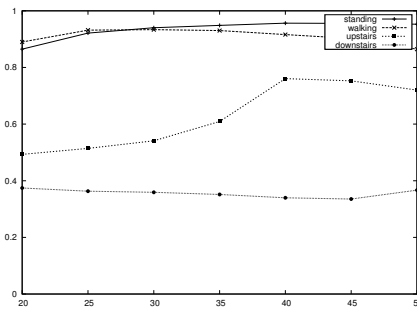


図 12 WSのみ変動させたときの適合率
Fig. 12 Precision in varying only WS.

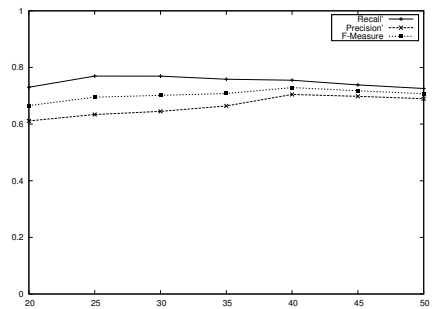


図 13 WSのみ変動させたときのF値
Fig. 13 F-Measure in varying only WS.

降りの再現率、適合率を両立させることは難しい。特に昇りの適合率と降りの適合率は、トレードオフの傾向が顕著であるため、適合率を重視するユーザが、昇り、降りの両方を自己抑制したい場合、現状のシステムでは実現が難しい。しかし、表3からわかる通り、F値が最大となるときの再現率は昇り、降り共に一定の水準を保っているため、再現率を重視するユーザであれば、アプリケーションとして提供できる。

3.3 ウィンドウサイズ (WS) のみ変動

次に、WSのみ変動させる実験を行った。ウィンドウは加速度値を格納する配列であり、WSが小さいほど古い加速度値を格納しなくなるため、最大値、最小値の更新頻度が多くなり、コンテキストの切り替わりが速くなる。しかし、WSが小さすぎると、例えば昇りのアクションをしているときに、加速度値が正から負に転ずる部分では一時的に最大値と最小値の絶対値が小さくなるため、昇り→歩行→降り→静止→昇りなどと細切れな認識をしてしまう恐れがある。そこでWSが与える影響を調べるため、WSのみを変動させたときの各コンテキスト毎の再現率を図11に、適合率を図12に、重み付けした再現率、適合率及びF値を図13にそれぞれ示す。

まず静止に関しては、図11から、WSが小さいほど再現率が高くなるのがわかる。これは、歩行→静止への切り替わりが速くなるからだと考えられる。一方、図12からWSが大きいほど適合率が上がっていることがわかるが、これは再現率とのトレードオフの関係が影響していると考えられる。

次に歩行に関しては、図11から、WSが小さいほど歩行の再現率が上がっていることがわかる。この理由には、階段→踊り場のときに切り替わりが遅くなることで、踊り場に入ったばかりの部分を昇り、または降りと誤認識することが考えられる。一方、適合率はトレードオフの関係からWSが大きいほど適合率も上がると考えられるが、図12からわかるように、実際はあまり変化が見られなかった。これは、踊り場→階段のときに切り替わりが遅くなることで、階段に入ったばかりの部分を歩行と誤認識をして、歩行の適合率が下がることが影響していると考えられる。つまり、トレードオフと切り替わりの遅さの影響が互いに均衡しているため、全体的に適合率の変化が見られなかったと考えられる。

昇りの再現率に関しては、図11からわかるように、WSが20から25に変動すると大きく再現率が増加し、25以上ではあまり変化しない結果となった。理由として、WSが20のときは、静止と歩行の誤認識による細切れが起き易いため再現率が低くなるが、WSが25以上では細切れが起き難くなるため、再現率が高くなったと考えられる。一方、適合率に関しても同様に、図12から、WSが大きい方が適合率が高くなる傾向であることがわかる。これは、WSが大きくなるにつれて階段では、昇り→歩行→降りのような細切れした認識が減ることで、途中で降りと誤認識することも減るからであると考えられる。

降りに関しては、図11からわかるように、WSが大きいほど再現率が上がっているが、これは昇りと同様に、細切れが起き難くなるからであると考えられる。また、図12が示す通り、WSを変動させても、適合率にはあまり影響を与えなかった。

最後に重み付けした再現率、及び、適合率、これらを統合したF値については、図13からわかるように、WSによる大きな変化は見られなかった。つまり、WSのパラメータ変動は、全体として再現率・適合率を観察した場合、あまり影響を与えないことが窺える。

4. まとめと今後の課題

本稿では、階段における歩きスマホを検知するため、ユーザの行動から得られた加速度値、及び、4つの閾値とウィンドウサイズ (WS) をパラメータとして、「静止」、「歩行」、「階段昇り」、「階段降り」の4つのコンテキスト認識を行うシステムを提案した。また、パラメータ最適化後の各コンテキストにおける再現率・適合率を明らかにし、再現率・適合率に与える各パラメータの依存性を調べた。その結果、閾値とWSのパラメータ変動では、全コンテキストの再現率・適合率を同時に高めることは難しいが、逐次パラメータを変動させることで、ユーザが自己抑制したいコンテキストに合わせて調整できる可能性があることを確認した。

階段における昇り降りを認識している従来研究には、本稿と同様に加速度センサのみを使用している文献[2],[3]と、加速度センサと気圧センサの両方を使用している文献[4],[5]がある。文献[2],[3]は機械学習を行うことで、パーソナライズされた認識を可能とするが、閾値を設けていないため、閾値の変動によって一部のコンテキスト認識に特化させることができない。一方、文献[4],[5]に使用されている気圧センサは、歩行、昇り、降りの区別を容易にするが、端末への搭載が少なくなってきているため実用的でない。また、気圧センサから得られる気圧値は、屋外の場合なら天気や気温によって変わり、屋内の場合なら天井の高さによって変わるので、様々な状況に応じた実装が必要になるなど、考慮しなければならぬ外的要因が増える。

今後の課題として、上記に挙げた従来研究は、いずれも階段昇降の認識のみであるが、混雑した階段静止(空間×アクション×周囲オブジェクト)などの認識も必要であり、カメラやNFCなどの別のセンサも組み合わせる。また今回は、第1著者1人のデータに対して最適なパラメータを求めたが、今後はこのパラメータを他の人間にも適用し、認識結果の検証を行っていく。さらに、鉛直方向における加速度値だけではなく、水平方向における加速度値の利用により、再現率・適合率を向上できるかどうかの検証も行っていく。

文 献

- [1] K. Yoshida, and S. Hattori, "Web Self-Controller: A Filtering Browser for Self-Control," ESKM 2013, pp.26-31 (2013).
- [2] T. Iso, and K. Yamazaki, "Gait Analyzer Based on a Cell Phone with a Single Three-axis Accelerometer," MobileHCI 2006, pp.141-144 (2006).
- [3] N. Ravi, N. Dandekar, P. Mysore, and M. L. Littman, "Activity Recognition from Accelerometer Data," AAAI 2005, pp.1541-1546 (2005).
- [4] 小川晶子, 小西勇介, 柴崎亮介, "自立型ポジショニングシステム構築に向けて～着用型センサーを用いた人間の行動モード推定～," 全国測量技術大会 2002 学生フォーラム発表論文集 (2002).
- [5] 森岡英之, 新津善弘, "複数センサを用いたコンテキスト推定法の検討," IPSJ 2010, 1ZD-2, pp.3-475-476 (2010).